NONPROVISIONAL PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

OLIFF & BERRIDGE, PLC
P.O. Box 19928
Alexandria, Virginia 22320
Telephone: (703) 836-6400
Facsimile: (703) 836-2787

Attorney Docket No.: 104324

Date: January 19, 2000

jc598 U.S. PTO 09/487583 01/19/00

BOX PATENT APPLICATION

NONPROVISIONAL APPLICATION TRANSMITTAL
RULE §1.53(b)

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

Transmitted herewith for filing under 37 C.F.R. §1.53(b) is the nonprovisional patent application

For (Title): METHODS FOR GENERATING ANTI-ALIASED TEXT AND LINE GRAPHICS IN COMPRESSED DOCUMENT IMAGES

By (Inventors): Dan S. BLOOMBERG and Luc VINCENT

☒ Formal drawings (Figs. 1-33; 17 sheets) are attached.
☒ A Declaration and Power of Attorney is filed herewith.
☒ An assignment of the invention to XEROX CORPORATION is filed herewith.
☐ An Information Disclosure Statement is filed herewith.
☐ A Preliminary Amendment is filed herewith.
☐ Please amend the specification by inserting before the first line the sentence --This nonprovisional application claims the benefit of U.S. Provisional Application No. _____, filed _____.--
☒ The filing fee is calculated below:

CLAIMS IN THE APPLICATION AFTER ENTRY OF
ANY PRELIMINARY AMENDMENT NOTED ABOVE

FOR:	NO. FILED	NO. EXTRA	RATE	FEE
BASIC FEE				$ 690
TOTAL CLAIMS	59 - 20	= 39*	x 18	$ 702
INDEP CLAIMS	1 - 3	= 0*	x 78	$
☐ MULTIPLE DEPENDENT CLAIMS PRESENTED			+260	$
* If the difference is less than zero, enter "0".			TOTAL	$ 1392

☒ Please charge Deposit Account No. 24-0037 in the amount of $1392. Two duplicate copies of this sheet are enclosed.
☒ The Commissioner is hereby authorized to charge any other fees that may be required to complete this filing, or to credit any overpayment, to Deposit Account No. 24-0037. Two duplicate copies of this sheet are attached.

Respectfully submitted,

James A. Oliff
Registration No. 27,075

Christine M. Hoeft
Registration No. 41,844

JAO:CMH/epb

Inventor Information

Inventor One Given Name::	Dan S.
Family Name::	BLOOMBERG
Name Suffix::	
City of Residence::	Palo Alto
State or Prov. of Residence::	California
Country of Residence::	USA
Inventor Two Given Name::	Luc
Family Name::	VINCENT
Name Suffix::	
City of Residence::	Palo Alto
State or Prov. of Residence::	California
Country of Residence::	USA
Inventor Three Given Name::	
Family Name::	
Name Suffix::	
City of Residence::	
State or Prov. of Residence::	
Country of Residence::	
Inventor Four Given Name::	
Family Name::	
Name Suffix::	
City of Residence::	
State or Prov. of Residence::	
Country of Residence::	
Inventor Five Given Name ::	
Family Name::	
Name Suffix::	
City of Residence::	
State or Prov. of Residence::	
Country of Residence::	

Correspondence Information

Name Line One::	Oliff & Berridge PLC
Address Line One::	P.O. Box 19928
City::	Alexandria
State or Province::	VA
Postal or Zip Code::	22320
Telephone::	(703) 836-6400
Fax::	(703) 836-2787
Electronic Mail::	commcenter@oliff.com

Application Information

Title Line One::	METHODS FOR GENERATING ANTI-ALIASED
Title Line Two::	TEXT AND LINE GRAPHICS IN
Title Line Three::	COMPRESSED DOCUMENT IMAGES
Title Line Four::	

Total Drawing Sheets:: 17
Docket Number:: 104324

Continuity Information

>This application is a::
Application One::
Filing Date::
Patent Number::
which is a::
>>Application Two::
Filing Date::
Patent Number::

Prior Foreign Applications

Foreign Application One::
Filing Date::
Country::
Priority Claimed::
Foreign Application Two::
Filing Date::
Country::
Priority Claimed::
Foreign Application Three::
Filing Date::
Country::
Priority Claimed::

Doc Code: AIA

METHODS FOR GENERATING ANTI-ALIASED TEXT AND LINE GRAPHICS IN COMPRESSED DOCUMENT IMAGES

BACKGROUND OF THE INVENTION

1. Field of Invention

This invention relates to a method and system for storing and generating anti-aliased text or lineart from compressed document image files. More specifically, the invention uses a Mixed Raster Content (MRC) model that represents the image as an ordered set or mask/image pairs at resolutions appropriate to the content of each layer. When using token compression, use of anti-aliased text or lineart improves text and lineart image appearance for both low and high resolution by smoothing edges and avoiding token baseline jitter.

2. Description of Related Art

Uncompressed grayscale or color scanned document images contain too much data for convenient on-line storage and retrieval. Lossless compression of a 300 ppi grayscale scanned image, using universal compression such as Lempel-Ziv, typically causes only a small reduction in stored data. Thus, for example, an 8 MB uncompressed image may be minimally shrunk to 4 to 7 MB after lossless compression. Compression is only minimal because most of the image data is produced as a result of scanner noise in the 3 or 4 least significant bits. Thus, some lossy compression is necessary. However, due to conflicting application requirements, there is no universal method that will fit all situations.

For example, suppose the requirement is that compression must be visually lossless. The amount of achievable compression is limited, and depends strongly on the scanning resolution. For example, at 300 ppi, simple hierarchical vector quantization (HVQ) provides a guaranteed 4x compression, with perhaps a 7x typical compression after further Lempel-Ziv coding. However, even at 8x compression, grayscale images are produced that compress to about 1.0 MB/page, which is too much for many applications.

To get a reasonable (but not lossless) image, at significantly better compression, a MRC approach may be used, in which the image is stored as ordered pairs of (mask,

image) layers. Mixed Raster Content (MRC) is one approach to satisfy the compression needs of differing types of data. MRC involves separating a composite image into a plurality of masks, and separately applying an appropriate compression technique to each image mask. The document is represented by a pixel map that is decomposed into a multiple mask representation.

The masks allow the image to be painted through, and the ordering is necessary because the last pixel painted in each location is the one that is apparent to a viewer. In the most simple but non-trivial example, two (mask, image) pairs are used. The first layer is the background image, represented as a low-resolution gray or color image, and its mask is taken to cover the entire image. The second layer is the text/lineart layer, represented by a binary high-resolution, e.g., a 300 ppi or greater mask and a very low resolution foreground color image that is painted through the high resolution mask. The foreground color image can be at even lower resolution than 100 ppi.

It is possible to conform with the MRC format, and use a 300 ppi text or lineart mask, compressed lossily using connected component tokens, and a 100 ppi background image compressed with JPEG or wavelets. It is also possible to use a third (mask, image) layer pair for higher resolution embedded color images that are located by a segmentor. This third image layer may also be compressed using JPEG or wavelets. A similar approach has also been used where text or lineart is also compressed lossily using binary image tokens and wavelet compression is used on the background image.

For these MRC formats, the text or lineart mask can also be compressed losslessly using Group4, Limpel-Ziv, or arithmetic coding. However, there are several problems associated with the current use of a binary text or lineart mask. First, regardless of the compression method used on the text or lineart mask, the text or lineart, when rendered, has stair-steps on nearly horizontal or vertical lines. The text or lineart image quality suffers from severe aliasing when sub-sampled. The poor quality is also evident when viewed at a higher resolution on a cathode ray tube (CRT), where the character boundaries display the noisiness of individual pixels. Another weakness of the conventional method is the baseline jitter problem when connected component tokens are used. It is very difficult to avoid visible baseline jitter when tokens are substituted,

because the vertical alignment of individual characters is susceptible to the noise on character boundaries introduced by thresholding the grayscale character image to binary.

These image quality deficits, stemming from the binary character of the text or lineart, make the conventional MRC format unacceptable for applications requiring that the compression loss is not observable, such as for a bookscanner where a visually lossless archival gray image must be saved.

SUMMARY OF THE INVENTION

The exemplary embodiments of the present invention ameliorate or significantly resolve these problems with conventional methods and systems of storing and generating image data in rendering apparatuses at a relatively small cost in additional storage. Such problems can be rectified to some extent, and at some cost, by allowing use of a grayscale text or lineart mask. This can be done within the MRC format in a number of ways. Alternatively, another (mask, image) pair may be added for the boundary pixels, i.e., a pixel that has at least one of its eight nearest neighbors being of a color opposite to the pixel's color, of the text or lineart mask. Additionally, methods to render these boundary pixels, e.g., without using an explicit layer pair, may be implemented. Such solutions can be made with only a relatively small incremental cost in storage.

The present invention provides a system and method for storing and generating anti-aliased text or lineart from document image files to improve the appearance of rendered text or lineart at both low and high resolutions. Each exemplary embodiment of the invention achieves the goals with minimal cost in additional storage and rendering complexity.

The exemplary embodiments of the invention generate an anti-aliased grayscale version of a text or lineart mask used within an ordered set of mask/image pairs. The grayscale data can be stored with either lossless compression or with lossy compression.

The exemplary embodiments of the present invention may also be applied to methods for anti-aliasing compressed representations of color images. To interpolate between the low resolution foreground and background color images, three values , e.g., for red, blue and green color separations, must be available for each boundary pixel.

All exemplary embodiments involve either generating or saving pixel values for pixels near the boundaries of the text or lineart mask, and using these values in the rendering process to smooth the visual appearance of the boundaries. If the data is compressed, the compressed representation for the pixels associated with the text or lineart mask and its boundary can be either saved as a set of full page images, or as a set of tokens that typically represent connected components in the foreground image.

The exemplary embodiments of the present invention separately provide improved display and print rendering of MRC-compressed images.

The exemplary embodiments of the present invention separately provide means for providing anti-aliased text or lineart data while minimizing costs of additional storage and rendering complexity.

It is preferred that the exemplary embodiments of the invention are implemented in conjunction with the processing of images from documents that contain different types of data, e.g., MRC.

The first exemplary embodiment uses high resolution grayscale scans. The second through fourth embodiments may be implemented with a combination of a high resolution binary scan and a low resolution grayscale scan. The fifth embodiment uses very high resolution binary scans.

The first through fourth embodiments provide methods that use an explicit or implicit extra MRC mask/image pair for the boundary pixels of the text or lineart mask. Methods according to the fifth embodiment generate the gray boundary pixels at high resolution directly from a very high resolution binary representation of the text or lineart mask.

The second and third exemplary embodiments classify the boundary pixels of each type (ON and OFF) depending on their neighboring pixels in the text or lineart mask image, and substitute a specific pixel value for each type globally. Additionally, second exemplary embodiment uses an adaptive algorithm that computes the best pixel values from the image; the other uses pre-computed pixel values for similar images. Additional storage for this method is insignificant or nil.

The third and fourth embodiments provide methods that can be used with current scanners, e.g., high resolution is 300 ppi binary, and current MRC formats, without any additional storage requirements.

The fourth method does not use information about the neighborhood of the boundary pixel (in the mask), but instead just substitutes a single global value for boundary pixels of each type.

The fifth exemplary embodiment substitutes the original pixels directly into the boundary. The quality and compressibility are controlled by the quantization applied to the stored pixels. The fifth method requires a very high resolution binary scan to generate the text or lineart mask at 600 ppi or greater. It then generates the anti-aliasing boundary values at 300 ppi by filtering and down-sampling. One of the simplest methods for doing so is the scale-to-gray conversion.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other objects of the present invention will be readily appreciated and understood from consideration of the following detailed description of exemplary embodiments of this invention, when taken with the accompanying drawings, in which:

Fig. 1 shows a high resolution original scanned image;

Fig. 2 shows the standard MRC version of the image shown in Fig. 1;

Fig. 3 shows an original scanned image after 4x grayscale sub-sampling;

Fig. 4 shows a cleaned MRC version of the image shown in Fig. 3 with binary mask after 4x sub-sampling;

Fig. 5 shows a schematic illustration of an image rendering system that implements each of the first through fifth exemplary embodiments;

Fig. 6 shows a flowchart outlining a method of image rendering in accordance with the first exemplary embodiment;

Fig. 7 shows an image rendered using a binary mask;

Figs. 8-11 show a set of images, corresponding to Fig. 7, rendered using selection of original pixels on the boundary, quantized to two, three, four and five bits in accordance with the first exemplary embodiment;

Fig. 12 shows an original image, as scanned, using all eight bits, and corresponding to Figs. 8-12;

Figs. 13-15 show rendering results where the boundary pixels are quantized to two, three and four most significant bits, respectively;

Fig. 16 shows a flowchart outlining a method of image rendering in accordance with the second exemplary embodiment;

Fig. 17 shows a chart of histograms of interior boundary pixels for, 0, 1, 2 and 3 four-neighbors of opposite (OFF) color;

Fig. 18 shows a chart of histograms of exterior boundary pixels for, 0, 1, 2 and 3 four-neighbors of opposite (ON) color;

Fig. 19 shows a chart of the median pixel values found from the unsmoothed histograms for the interior boundary pixels;

Fig. 20 shows a chart of the median pixel values found from the unsmoothed histograms for the exterior boundary pixels;

Fig. 21 shows a magnified text fragment that is produced by substituting the mean values shown in Figs. 19 and 20 globally for the boundary pixels using 4-connected median adapted pixel values in accordance with the second exemplary embodiment;

Fig. 22 shows the result of the 4x sub-sampled image corresponding to Fig. 21;

Fig. 23 shows the magnified text fragment that is produced by substituting the mean values shown in Figs. 19 and 20 globally for the boundary pixels using 8-connected median adapted pixel values in accordance with the second exemplary embodiment;

Fig. 24 shows the result of the 4x sub-sampled image corresponding to Fig. 23;

Fig. 25 shows a flowchart outlining a method of image rendering in accordance with the third exemplary embodiment;

Fig. 26 shows the magnified text fragment that is produced with a grayscale mask boundary using a non-adapted linearly interpolated approximation to 4-connected median pixel values in accordance with the third exemplary embodiment;

Fig. 27 shows the result of the 4x sub-sampled image corresponding Fig. 26;

Fig. 28 shows a flowchart outlining a method of image rendering in accordance with the fourth exemplary embodiment;

Fig. 29 shows the magnified text fragment that is produced with all pixels of the interior boundary set to a single threshold-dependent value, and all pixels of the exterior boundary set to another threshold-dependent value in accordance with the fourth exemplary embodiment;

Fig. 30 shows the result of the 4x sub-sampled image corresponding Fig. 29;

Fig. 31 shows a flowchart outlining a method of image rendering in accordance with the fifth exemplary embodiment;

Fig. 32 shows the magnified text fragment that is produced with all pixels of the interior boundary set to a single threshold-dependent value, and all pixels of the exterior boundary set to another threshold-dependent value in accordance with the fourth exemplary embodiment; and

Fig. 33 shows the result of the 4x sub-sampled image corresponding Fig. 29.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

Each of the exemplary embodiments of the invention assigns gray values to the boundary pixels of the text or lineart mask. Each of these exemplary embodiments provides methods for grayscale anti-aliasing that are extended to color images, for text or lineart masks that are compressed either as images or as tokens. It is foreseeable that color rendering of anti-aliased tokens will have a significant affect on scanning devices.

Presently, scanning systems typically perform a 300 ppi binary scan for high resolution foreground components, and a 100 ppi color scan for foreground and background color images. The present invention provides the capability of using a grayscale value for the boundary pixel to render, by interpolation, color pixels when the text or lineart mask is compressed either as a full image or in tokens, and using any of the five exemplary embodiments of the invention.

The present invention is of use in any situation where it is desirable or necessary to render the text and lineart at greater than 300 binary ppi in an image. This is of particular use when printing, to eliminate obvious stair-steps in lines that are nearly horizontal or vertical, when viewing a magnified image, e.g., a 300 ppi image on an 80

ppi CRT, or when viewing a highly sub-sampled image, as when attempting to image an entire page scan in a window on a CRT display. When lossy token compression is used for the text or lineart mask, implementation of the exemplary embodiments also improves the appearance of printed and displayed text or lineart by reducing the baseline jitter that is difficult to handle satisfactorily when using binary tokens at 300 - 400 ppi.

It is now recognized that a MRC image architecture is flexible enough to provide high image quality with excellent compression, and standardization efforts are under way. The present invention fits well into the MRC architecture. As explained below, some of the exemplary embodiments utilize an additional image layer, others rely on the use of higher resolution binary masks; others provide an improvement in rendering from current MRC compressed images, altering either the specific set of MRC layers or the data that is stored in them.

Exemplary embodiments of the invention may be used in conjunction with a 300 ppi grayscale scan and a 100 ppi color scan. The high resolution scan, i.e., 300 ppi, is performed in grayscale because this permits the use of adaptive thresholding to generate the binarized text or lineart mask. If instead a binary high resolution scan is used, where the scanner has performed a global thresholding, the text or lineart mask can be flawed in regions where the background is incorrectly thresholded. Thus, it is preferred that a file format has the high resolution grayscale values for boundary pixels available.

It is preferred that the file format for storing the foreground mask as tokens is extensible. In addition to a token index and location, which are currently saved, the color, character and font ID, and anti-aliasing information can also be stored with each token instance.

The importance of anti-aliasing is most evident for document images viewed on a display. Without anti-aliasing, at very high resolution, such as the full resolution of the original binary scan, edge pixellation effects are quite noticeable. And at low resolution, e.g., 3-4x sub-sampling, the aliasing due to sub-sampling of a binary mask severely compromises legibility of small text.

Therefore, the present invention is directed at solving problems of aliasing, noisy characters, and baseline token jitter described above. Accordingly, the system and

method allow images to be displayed at both high and low (sub-sampled) resolution, without the defects evident from thresholding. The exemplary embodiments of the invention provide a variety of approaches for achieving this capability in the context of a layered MRC format. The exemplary embodiments provide improved visual quality, achievable compression, ease of implementation, and speed of decompression. Therefore, the exemplary embodiments achieve improved rendering.

In the detailed explanation of the exemplary embodiments, high resolution data means a 300 - 400 ppi binary or grayscale image. Very high resolution binary data refers to a binary image at 600 ppi or greater.

Images can be transformed between high resolution gray and very high resolution binary, and vice-versa. In the up-resolution direction, i.e., high gray resolution to very high binary resolution, an interpolative gray-to-gray expansion is followed by thresholding to binary. In the down-resolution direction, i.e., very high binary resolution to high gray resolution, a scale-to-gray conversion is used, which is typically a low-pass filter followed by sub-sampling.

To implement some of the exemplary embodiments, e.g., the first and fifth embodiments, it is necessary to acquire either a very high resolution binary scanned image (fifth embodiment) or a high resolution grayscale scanned image (first embodiment) with at least 2 bits per pixel (bpp) and preferably 4 bpp. Because these images can be interconverted, each of the first and fifth embodiments requires either a high resolution gray or very high resolution binary image because the method and system can acquire the necessary data from either type of image data. For example, if a method requires a very high resolution binary image, it can directly obtain the image data or obtain it from a scanned high resolution gray image.

However, the second through fourth embodiments do not require anything beyond standard high resolution binary scanned image data.

All embodiments generate or save pixel values for pixels near the boundaries of the text or lineart mask, and use these values in the rendering process to smooth the visual appearance of the boundaries. There is no need to explicitly store a mask that identifies

the boundary pixels of the text or lineart mask, because these are easily derived from the text or lineart mask itself.

A compressed representation for pixels associated with a text or lineart mask and its boundary can either be saved as a set of full page images, or as a set of tokens that typically represent connected components in the foreground image. Although specific factors affect anti-aliasing a token representation of a text or lineart mask, the method of compressing the text or lineart mask does not conceptually affect the image quality.

During description of the five exemplary embodiments, various experiments were developed to provide examples to exemplify the operation and resulting benefits of the embodiments. All examples relate to an image of text in 8 pt. font, that was scanned at 300 ppi and 8 bpp. High resolution display examples use a single word of the text. Low resolution examples use a larger block consisting of about 10 lines of text. High resolution display examples are magnified about 7x, relative to the size they would have if rendered at a scan resolution on a 300 ppi printer. At this resolution, the image data are about twice as big as they would appear at full (300 ppi) resolution on an 80 ppi monitor. The low resolution images are derived by sub-sampling the full resolution image by 4x, and then printing it at a size roughly comparable to both the size on the original scanned page and the direct display on an 80 ppi screen. Each pixel in the sub-sampled image is thus rendered as approximately a 4x4 block of halftoned printed pixels.

In the experiments implementing MRC, a globally adapted threshold value (152) is used for extracting the binary text or lineart mask. The method sub-samples pixels from the image at a specified rate, e.g., 8x, and forms a histogram of pixel values with overlapping bins. The lightest bin to have more than a given fraction of all the pixels, e.g. 15 %, sets the background value. This background value is adjusted downward to determine the global threshold to be used.

Once the text or lineart mask has been identified, the text or lineart mask boundary pixels are brought up to the general background level using a small (5x5) grayscale dilation. This grows the lighter background pixels through the boundary pixels, and it also lightens and smoothes the background. Superposing the text or lineart mask

on this smoothed background gives the rendered MRC image, as it is currently done with a binary text or lineart mask.

Figure 1 shows the high resolution original scanned image and Figure 2 shows the standard MRC version. If the original image was sub-sampled 4x and displayed on an 80 ppi monitor, the result is barely legible. Figure 3 shows in print approximately what would be seen on a display screen. However, when the standard MRC image with binarized mask is sub-sampled 4x, as shown in Figure 4, the results are not legible.

In each of the exemplary embodiments except for the fifth embodiment the text or lineart mask is decomposed into two parts: interior and boundary. The interior part can be constructed from the binary text or lineart mask, by removing the ON boundary pixels, i.e., those ON pixels that are touching an OFF pixel. A morphological erosion of the single mask using a 3x3 structuring element gives a simple implementation. However, the boundary part is composed of all boundary pixels to the mask, both ON and OFF; namely, any pixel in the text or lineart mask that is adjacent, in an 8-connected sense, to a pixel of the opposite value. It is the union of (1) the pixels removed in constructing the interior part of the text or lineart mask and (2) the OFF pixels in the background that are adjacent to those boundary pixels. An implementation for construction of the boundary part performs the exclusive or operation, "XOR", of the erosion and dilation of the text or lineart mask, with both morphological operations using a 3x3 structuring element.

The first through fourth exemplary embodiments set the interior pixels to black. However, each exemplary embodiment differs in how it handles the boundary pixels.

Figure 5 illustrates a schematic diagram of an image rendering system 500 implementing the exemplary embodiments of the present invention. Each of the exemplary embodiments may be implemented within the image rendering system 500. As shown in Fig. 5, the image rendering system 500 includes a scanner 510, an encoder 520, a decoder 530, memory 540. The scanner 510 is coupled to the encoder 520 and provides scanned image data to the encoder 520. The encoder 520 is also coupled to the memory 540. The memory 540 is also coupled to the decoder 530. The decoder may be incorporated in an image renderer, for example, a print engine with, e.g., a

conventionally configured flying spot raster output scanner , an image rendering device for displaying an image on a display screen, etc.

In accordance with a first exemplary embodiment, the scanner 510 scans an image to provide image data. The image data is provided to the encoder 520 that encodes the image data. Such encoding may include compression or other beneficial data manipulation. Encoding begins by identifying boundary pixels within the image data provided by the scanner 510. These boundary pixels, i.e., the actual scanned pixels on the boundary of the text or lineart mask, appropriately quantized, are saved in memory 540 by the encoder 520 for use subsequently in rendering the image.

In accordance with the first exemplary embodiment of the invention, anti-aliasing of the text or lineart data begins with a special scanning scheme, e.g., either binary very high resolution or grayscale high resolution, used to produce high resolution grayscale data, e.g., 300 - 400 ppi. Encoding is not performed in an adaptive manner. Therefore, no global pixel value assignments are computed from the scanned image. There is no connectivity utilized. Therefore, the values of connected neighbors of a boundary mask pixel are not used in determining the pixel value assignments encoded data that is stored in memory 540. The individual derivation of grayscale boundary pixels is consistent with MRC in that the boundary pixels constitute a (mask, image) pair, where the mask is just the boundary mask described above, and the image consists of 0s (or 1s) everywhere except under the boundary mask, where the pixel value represents the actual value, quantized to some number of significant bits. The boundary pair resulting from encoding according to the first exemplary embodiment is explicit. Therefore storage of an extra MRC boundary mask/image pair in memory 540 is required to subsequently render the image.

Figure 6 shows a flowchart indicating the flow of a method for providing anti-aliased text and lineart data in accordance with the first exemplary embodiment. The method begins in step S600 and control proceeds to step S610. In step S610, a special scan is performed to generate the grayscale image data at high resolution. Control then proceeds to step S620. In step S620, the grayscale image data is binarized to produce high resolution binary data using a set threshold. Control then proceeds to step S630, in

which the boundary pixels of the binarized image data are identified and control proceeds to step S640. In step S640, the grayscale values of the boundary pixels identified in step S630 are stored and control proceeds to step S650. In step S650, the high resolution binary pixel data and the stored grayscale values of the boundary pixels are stored. Control then continues to step S660. In step S660, the high resolution binary pixel data and the grayscale values of the boundary pixels values are used to render the image and control proceeds to step S670. In step S670, the method ends.

During individual derivation of grayscale boundary pixels, the more bits retained, the more similar a subsequently rendered image will be to the original, scanned image, and the more expensive the storage. To illustrate, Figures 7-12 show the rendering results with some magnification, as in Figs. 1 and 2. Fig. 7 shows an image rendered using a binary mask on cleaned background. Figs. 8-11 show an image rendered in which the boundary pixels are displayed, quantized to two, three, four and five most significant bits, respectively. Fig. 12 shows the original scanned image.

It is also instructive to see the images of text when treated this way and then subsampled by 4x. Figs. 13-15 show rendering results where the boundary pixels are quantized to two, three and four most significant bits, respectively. Comparing with the use of a binary mask in Fig. 7, it is evident that anti-aliasing is provided by rendering the boundary region in gray tones. Further, much of the anti-aliasing effect is provided by the two most significant bits. When the image is rendered on a display screen, the difference amongst these three images is even more apparent, but even there, the use of three and four most significant bits give diminishing returns.

In the data file holding the image data, only the original single foreground mask needs to be stored because the interior and boundary masks are easily derived from it, as described above. The boundary pixels, when quantized to two or three bits, are stored in a full boundary image, where all non-boundary pixels are given the value zero (or max). This image should compress reasonably well using Lempel-Ziv. Alternatively, since the boundary pixel locations are known from the mask, it is not necessary to store the boundary pixels within a full image. Instead, the boundary pixels can be stored in a

packed sequential array, in raster scan order, and their location in the image can be inferred from the foreground mask.

Figure 16 shows a flowchart indicating the flow of a method for providing anti-aliased text and lineart data in accordance with the second exemplary embodiment. The method begins in step S1600 and control proceeds to step S1610. In step S1610, the image is scanned specially to determine the actual grayscale values of the boundary pixels at high resolution. Control then proceeds to step S1620. In step S1620, the grayscale image data is binarized to produce high resolution binary data using a set threshold. Control then proceeds to step S1630. In step S1630, the boundary pixels are identified and control proceeds to step S1640. In step S1640, the boundary pixels are separated into interior and exterior pixels, which are boundary pixels that are ON and OFF, respectively, in the text or lineart mask. Control then proceeds to step S1650, in which each subset, e.g., interior and exterior pixels, are analyzed separately to determine connectedness to determine an appropriate gray value for each of the boundary pixels. Performance of step S1650 is based on the assumption that the grayscale value of the original boundary pixel is correlated with the number of connected pixels in the mask image of opposite (or same) color. Subsequently, control proceeds to step S1660, gray scale values are calculated for the boundary pixels and control proceeds to step S1670. In step S1670, the high resolution binary boundary pixel values and the derived grayscale values of the grayscale pixel data are stored in memory as compressed data. Control then proceeds to step S1680 in which the high resolution binary boundary pixel data and the derived grayscale values of the boundary pixels are used to render the image and control proceeds to step S1690. In step S1690, the method ends.

Two pixels are 4-connected if their positional relation is such that a second pixel is positioned adjacent to a first pixel to the immediate left, right, top or bottom of the first pixel. Two pixels are 8-connected if their positional relation is such that the second pixel is positioned adjacent to the first pixel to the immediate left, right, top or bottom or to the immediate upper-left, upper-right, lower-left or lower-right of the first pixel. As mentioned above, an assumption is made that the grayscale value of the original boundary pixel is correlated with the number of connected pixels in the mask image of opposite (or

same) color. For example, an interior (ON) pixel with three 4-connected OFF pixels is expected to be lighter than one with only one 4-connected OFF pixels. Therefore, each of the boundary pixels of each type is analyzed in the image to form an estimate of its gray value from the connectivity. This estimate is formed by building a histogram of pixel values for each type and for each connectivity, and taking the median pixel value.

Step S1640 may be performed using either type-4 or type-8 connectivity. Experiments have been performed using both type-4 connectivity (analyzing the four orthogonal pixels surrounding a sampled pixel) and type-8 connectivity (analyzing the four orthogonal pixels as well as the four diagonal pixels surrounding the sampled pixel. The differences in results between type-4 and type-8 are not large. However, the type-4 connectivity evaluation results are discussed below in more detail because they are easier to visualize.

To illustrate the validity of the assumption of pixel value correlation with connectivity, Figures 17 and 18 show a plot of histograms for interior and exterior boundary pixels respectively. Fig. 17 shows the histograms of interior boundary pixels, for 0, 1, 2 and 3 four-neighbors of opposite (OFF) color. Figure 18 shows the histograms of exterior boundary pixels, for 0, 1, 2 and 3 four-neighbors of opposite (ON) color.

Four curves can be distinguished in each of Figs. 17 and 18. In Fig. 17, the interior histogram, these curves are for 0, 1, 2, and 3 connected pixels of opposite (OFF) color. There are few cases where four connected pixels are OFF, and during analysis these pixel values are set to the threshold value used for constructing the text or lineart mask. In Fig. 18, the exterior histogram, the four curves are again for 0, 1, 2 and 3 connected pixels of opposite (ON) color. There are few cases with pixels with four ON 4-neighbors. During analysis, these pixel values are also set to threshold value used for constructing the text or lineart mask. In the data shown in Figs. 17 and 18, the threshold value used for constructing the text or lineart mask was 152. As a result, it should be recognized that there are some pixels in each histogram that cross over the threshold that was used to generate the mask. This cross over is merely an artifact of the use of a smoothing filter of length 7 that was applied to the histograms.

The median pixel values are found from the (unsmoothed) histograms, and are shown for interior and exterior boundary pixels in Figures 19 and 20, respectively. In each of these Figures, the median pixel value is plotted against the number of mask neighbors of opposite color.

Figure 21 shows a magnified text fragment that is produced by substituting the mean values shown in Figs. 19 and 20 globally for the boundary pixels. Fig. 21 shows the MRC image with grayscale mask boundary using 4-connected median adapted pixel values. The magnified text fragment shown in Fig. 21 is noisier than the corresponding fragment of the original image but storing the image data only requires storing the ten median values in addition to the standard MRC data. Figure 22 shows the result of the 4x sub-sampled image.

Figure 23 also shows the magnified text fragment that is produced by substituting the mean values shown in Figs. 19 and 20 globally for the boundary pixels. However, Fig. 23 shows the MRC image with grayscale mask boundary using 8-connected median adapted pixel values. Figure 24 shows the result of the 4x sampled image. Comparison of Fig. 21 and Fig. 23 indicates that there is little apparent difference in image quality between the 4 and 8 connected versions, when viewed on a display.

In a third exemplary embodiment, non-adaptive, connectivity-dependent global pixel values are generated. If high resolution grayscale values are not available, a set of values may be derived from analysis on similar images. A standard scanning scheme, e.g., binary high resolution, is used to produce high resolution binary data, e.g., 300 - 400 ppi. The method is not adaptive. Therefore, global pixel value assignments are not computed from the scanned image. However, connectivity is utilized. Therefore, the values of connected neighbors of a boundary mask pixel are used in determining the pixel value assignments. The resulting boundary pair is implicit, which means the extra mask/image pair is constructed to render the image.

Figure 25 shows a flowchart indicating the flow of a method for providing anti-aliased text and lineart data in accordance with the third exemplary embodiment. The method begins in step S2500 and control proceeds to step S2510. In step S2510, a standard scan is performed to generate the binary image data at high resolution. Control

then proceeds to step S2520. In step S2520, boundary pixels are identified and control proceeds to step S2530. In step S2530, an estimate of the median pixel values on the boundaries is performed and control proceeds to step S2540. In step S2530, the estimate is performed based on the observation that, the median pixel value varies approximately linearly with the number of 4-neighbors of the opposite color. In step S2540, the boundary pixels are separated into interior and exterior pixels and control proceeds to step S2550. In step S2550, an approximate dependency on the number of oppositely colored 4-neighbors n is determined from the estimated median pixel values and control proceeds to step S2560. In step S2560, the grayscale values of the boundary pixels and the binary high resolution data are stored in memory. Control then proceeds to step S2570. In step S2570, the grayscale boundary pixel data and the binary high resolution data are used to render the image and control proceeds to step S2580. In step S2580, the method ends.

The third exemplary embodiment is particularly effective when it is not possible to generate a high-resolution grayscale image, from which pixels can either be selected or adapted based on the statistics of the image. In such a case, it is still possible to perform an estimate of the median pixel values on the boundaries, based on the observation that, when segregated by the number of 4-neighbors, the median pixel value varies approximately linearly with the number of 4-neighbors of opposite color. In the third embodiment, the boundary pixels are separated into interior and exterior pixels.

An approximate dependency of the boundary pixel gray value on the number of oppositely colored 4-neighbors n can be determined from median value curves, such as those in Figures 19 and 20, by making a linear fit to the data end points (for 0 and 4 neighbors). Therefore, for interior pixels, the pixel value for n = 0 is always assumed to be a fraction, for example, 0.45, of the threshold. Also, for n = 4, the pixel value is equal to the threshold. Hence, for all values of n,

$$val_{in} = 0.45T + (0.138T)n$$

where T is the threshold. Likewise, for exterior pixels, the pixel value for n = 0 is always taken to be equal to T plus a fixed fraction, for example, 0.47, of the difference between

255 and T, which reduces to 119 + 0.53T. Again for n = 4, the pixel value is T. Hence, the linear interpolation formula for all values of n is:

$$val_{out} = 119 + 0.53T + (0.117T - 30)n$$

When these values are used in place of the (unknown) actual median values, the image quality is reasonably good. Using this method, the magnified fragment is shown in Figure 26 and the 4x subsampled image is shown in Figure 27. Figure 26 specifically shows a magnified MRC image with a grayscale mask boundary using non-adapted linearly interpolated approximation to 4-connected median pixel values. Figure 27 shows a MRC image with a grayscale mask boundary using non-adapted linearly interpolated approximation to 4-connected median pixel values after 4x sampling.

In a fourth exemplary embodiment, non-adaptive, connectivity-dependent global boundary pixels are assigned. This is performed by assigning interior boundary pixels a specified global value, and assigning exterior boundary pixels a specified global value. These values can be determined based on the median values in the scanned image, or assigned based on analysis of similar images. A standard scanning scheme, e.g., binary high resolution, is used to produce high resolution binary data, e.g., 300 - 400 ppi. The method is not adaptive. Therefore, global pixel value assignments are not computed from the scanned image. Furthermore, connectivity is not utilized. Therefore, the values of connected neighbors of a boundary mask pixel are not used in determining the pixel value assignments. The resulting boundary pair is implicit, which means the extra mask/image pair is constructed to render the image.

Figure 28 shows a flowchart indicating the flow of a method for providing anti-aliased text and lineart data in accordance with the fourth exemplary embodiment. The method begins in step S2800 and control proceeds to step S2810. In step S2810, a standard scan is performed to generate the binary image data at high resolution. Control then proceeds to step S2820. In step S2820, the boundary pixels are identified and control proceeds to step S2830. In step S2830, the boundary pixels are separated into interior and exterior boundary pixels and control proceeds to step S2840. In step S2840, a first global grayscale value is set for the interior boundary pixels and a second global grayscale value is set for the exterior boundary pixels. Control then proceeds to step

S2850. In step S2850, the first global grayscale value is assigned to all interior boundary pixels and the second global grayscale value is assigned to all exterior boundary pixels. Control then proceeds to step S2860. In step S2860, the grayscale boundary pixel values assigned in step S2850 and the binary high resolution data provided in step S2810 are stored in memory as compressed data. Control then proceeds to S2870, in which the grayscale boundary pixel values and the binary high resolution data are stored in memory. Subsequently, control proceeds to step S2880, in which the grayscale boundary pixel values and the binary high resolution data are used to render the image and control proceeds to step S2890. In step S2890, the method ends.

Implementation of the fourth exemplary embodiment is performed when high resolution grayscale data is unavailable and simplification of the method for performing anti-aliasing of the text or lineart mask is of heightened importance, for example, in a rendering engine that does not have time or computational ability to determine the number of oppositely colored adjacent pixels for each boundary pixel It is worth noting that at some degree of simplification, rendering results are sufficiently poor so that it makes no sense to simplify further. In this exemplary embodiment, connectivity is ignored, a global value is assigned to all interior boundary pixels and another global value is assigned to all exterior boundary pixels. Therefore, only two parameters are chosen, and these global values can be taken as the median values of representative boundary pixels from other images in which the high resolution gray value is known. These values are found to be approximately $val_{in} = 0\ 62T$ and $val_{out} = 77 + 0.70T$. The former is determined assuming the median value scales with threshold T as aT, and the latter is determined assuming the median value scales with threshold as T + a(255 - T). These are the same forms as were used in the third exemplary embodiment for fitting the point n = 0.

The results are illustrated in Figure 29 for the magnified fragment and in Figure 32 for the 4x subsampled image. For these images, the interior boundary pixels are set to a threshold value of 95 arid the exterior ones are set to a threshold value of 183. The results are better than using a binary text or lineart mask, but worse than the other

methods. The advantage here is that it is not necessary to compute the connectivity of each pixel in the boundary.

In a fifth exemplary embodiment, a direct scale-to-gray is performed from a very high resolution (600 ppi or more) binary mask. A special scanning scheme, e.g., either binary very high resolution or grayscale-high resolution, is used to produce very high resolution binary data, e.g., 600 ppi or greater. The method is adaptive, and local boundary pixel value assignments are computed from the scanned image. The values of connected neighbors of a boundary mask pixel are not used in determining the pixel value assignments. Therefore, there is no need to use an extra mask/image pair to render the image.

The fifth exemplary embodiment of the invention is the only exemplary embodiment that does not clearly fit within the MRC framework. Figure 31 shows a flowchart indicating the flow of a method for providing anti-aliased text and lineart data in accordance with the fifth exemplary embodiment. The method begins in step S3100 and control proceeds to step S3110. In step S3110, very high resolution binary text or lineart mask data, of 600 ppi or greater, is obtained. The very high resolution binary text or lineart mask data can be obtained from a 300 ppi grayscale image, or scanned directly and control proceeds to step S3120. In step S3120, the text or lineart data is then filtered and control proceeds to step S3130. In step S3130, the filtered data is down-sampled to 300 ppi gray. Subsequently, control proceeds to step S3140, in which thresholding is performed on the high resolution grayscale pixel data to binarize the image data. Control then proceeds to step S3150, in which boundary pixels are identified using the binary image data produced in step S3140. Control then proceeds to step S3150, in which the high resolution binary image data and the grayscale boundary pixel data is stored in memory as compressed data. Control then proceeds to step S3170, in which the compressed data is used to render the image and control proceeds to step S3180. In step S3180, the method ends.

The first step --the reduction from very high resolution binary to high resolution gray-- is often called " scale-to-gray", and is typically done by tiling the image preparatory to subsampling, and calculating a gray value for each tile that is proportional

to the number of ON pixels in the tile. In this case, the filtering operation is to take a convolution with a constant filter of size equal to the tile. A number of very fast conventional algorithms have been developed for this operation. Additionally, other methods of generating the 300 ppi gray data are useful. For example, the very high resolution binary image may be generated by performing a 2x grayscale expansion on input 300 ppi grayscale image date to rescale from 300 ppi to 600 ppi. Subsequently, the usual thresholding would be performed to generate the very high resolution binary mask which is stored in the file format, and on which the scale-to-gay operation is performed as in this exemplary embodiment. To perform anti-alias boundary pixel rendering from this high resolution grayscale text or lineart mask, while also filtering and smoothing the background, the grayscale text or lineart mask is placed over the unfiltered original background, as follows: (1) lowpass the grayscale text or lineart mask with a very high threshold value, e.g., 250, so that all gray pixels associated with the mask are kept; (2) clean the background with the usual grayscale dilation; and (3) select the gray pixels from the grayscale mask that are under the lowpassed binary version of this mask.

The magnified image fragment resulting from rendering in accordance with the fifth embodiment is shown in Figure 32. Because the grayscale pixels form a boundary that is only one pixel wide, instead of the usual two-pixel grayscale boundary from the original and the other methods, the boundary edges appear to be sharper. For example, a 4x sub-sampled image is shown in Figure 33. It appears sharper than images anti-aliased by other means because of the narrow boundary region, and it is comparable legibility to the other methods. In addition to the lack of fuzziness, the boundary pixels have a limited dynamic range, of five levels for the example shown where a 600 ppi binary mask is reduced to a 300 ppi grayscale mask. The grayscale text or lineart mask is quantized to five levels, which requires little more than two bits of data, because a 2x sale-to-gray conversion is performed from a binary mask.

In the MRC format, the binary mask would be stored in a file. However, storage for the 600 ppi binary mask in the file scales approximately linearly with the resolution. Therefore storage requires about double the storage space of the usual 300 ppi text or lineart mask. Thus, for rendering, the stored mask is scaled to gray at the high resolution

(300 ppi), and all pixels that are not pure white are incorporated in the rendered image, as described above.

As mentioned above, the exemplary embodiments of the invention may be implemented in combination with lossy text or lineart mask compression by tokens. For example, the present invention may be implemented with lossy compression by connected component tokens. Conventionally, such tokens are binary images that are tiled to compose the text or lineart mask, i.e., the MRC foreground mask. Conventionally, these tokens, i.e., binary tokens, are generated from a binarized image, typically at 300 ppi.

A binary token is usually taken, in the unsupervised classification scheme, to be the first token instance found of a new class of connected components. It is possible to form a composite grayscale token, by suitably aligning and averaging the pixel values in all token instances of a particular class. However, this approach has several major problems: alignment is tricky; it is computationally intensive and if there is only one token instance, the token remains binary.

Use of grayscale tokens alleviates the token substitution baseline jitter problem because gray pixels around the boundary of the token reduce the visual effect of hard pixel boundaries that can vary by ± 1 pixel in the standard token replacement algorithm. It also alleviates the anti-aliasing problems on the text or lineart mask in general. Each grayscale token can be considered to have three regions: foreground (ON pixels), boundary (gray pixels) and the remaining (OFF) pixels. The foreground and boundary layer pixels are non-intersecting. For implementing some of the exemplary embodiments of the invention, it may be necessary to decompose the boundary layer into two parts: an interior part, e.g., ON pixels in the binary mask, and an exterior part, e.g., OFF pixels in the binary mask. In no case does the boundary need to be saved, because it is either defined implicitly as the set of boundary pixels in the text or lineart mask or it can be derived from the saved grayscale pixels. It is worth noting, that the grayscale boundary pixels, when saved, achieve most of their effect from the 2 or 3 most significant bits.

Typically, the binary tokens are substituted for the token instances using an alignment based on the bounding boxes of each. For example, the lower-left comer of the

token is placed at the lower-left corner of the bounding box of the token instance. Special efforts can be made during compression to align the binary token individually with each token instance in the image, based on pixel locations, e.g., by finding an alignment that minimizes the pixel-wise XOR operation or maximizes the pixel-wise AND operation performed on the two images, or that aligns the center of masses to the nearest pixel. However, if special alignment techniques are not performed, the resulting baseline jitter will be noticeable. This is because the bounding box for the set of token instances can be displaced one pixel up or down (but not both) relative to the best visual alignment during the thresholding process. Then successive tokens in a line of text or lineart can be vertically displaced from the best alignment by up to two pixels with respect to each other because one token can be displaced upwards and the next downwards.

A random set of relative displacements, with a maximum of two, is visible as baseline jitter. If the tokens are optimally aligned with each instance of the token, jitter is significantly reduced, but the maximum relative displacement is still about one pixel. When the tokens are rendered with grayscale anti-aliasing pixels on the boundary, the observable jitter is significantly reduced, whether or not each binary token has been optimally aligned with all of the token instances. It is foreseeable that the observable reduction in baseline jitter using anti-aliased tokens is about a factor of two, depending on the specific method of generating the anti-aliasing pixels. Thus, the maximum relative baseline jitter will be about one pixel for unaligned tokens and about 0.5 pixel for aligned tokens. The jitter in the former case may not be noticeable, and it will surely be unobservable in the latter.

Each of the exemplary embodiments described above may be implemented in conjunction with grayscale token compression. The changes from using full image masks are relatively small. Because the number of tokens will be limited, the extra storage required for these grayscale boundary pixels must be less than that required if the full mask is stored rather than the tokens.

In conjunction with the first exemplary embodiment, the boundary pixels are selected from the original image, appropriately quantized and each token analyzed as two separate layers: foreground and boundary. These layers constitute two images that can be

compressed separately. The foreground image is binary and consists of the standard text or lineart mask, eroded to remove the interior boundary pixels. However, if the order of rendering of the layers is specified and the foreground is rendered first and the boundary second, it is not strictly necessary to erode the foreground text or lineart mask because the interior boundary pixels will paint over them.

As explained above in the detailed explanation of the first exemplary embodiment, the boundary pixels in the token can be stored as either a full image or as a packed array of pixels that are just the boundary pixels encountered in a raster scan order. When stored as a full image, the boundary image is grayscale , with the non-boundary pixels taking on a constant value, e.g., zero. When compressed by a universal source coder, e.g., Lempel-Ziv, few bits will be allocated to the non-boundary pixels. Nevertheless, the packed array of boundary pixels will always compress better than the full image.

In conjunction with the second exemplary embodiment, the analysis of grayscale pixel values based on the value in the high resolution grayscale image and the nearest neighbor colors in the thresholded mask is performed as explained above. The grayscale tokens are rendered by analyzing the connectivity of the boundary pixels and making the appropriate grayscale substitutions. There is no need to store an explicit layer (representation) of the boundary pixels. Again, only nine or ten numbers need to be stored when 4- or 8-neighbor connectivity is used, and these grayscale values are used in all the tokens.

In conjunction with the third exemplary embodiment, the grayscale tokens are as explained above, except that the substitution values are known a priori, and are not determined from the image being compressed. Rather, the substitution values are specified depending on whether the pixel is in the interior or exterior boundary, and on its 4- or 8-neighbor connectivity. The binary text or lineart mask is saved, with no extra data, except perhaps for the threshold value that was used to generate the text or lineart mask. It is not necessary to do a high resolution grayscale scan.

In conjunction with the fourth exemplary embodiment, the boundary grayscale pixel values are assigned as explained above based on constants for the interior and

exterior boundary, independent of 4- or 8-neighbor connectivity. As in the implementation of the third exemplary embodiment with grayscale token compression, only the threshold value needs to be saved in the file, but the appearance of these tokens is visually inferior because each boundary layer is given a single color. It is not necessary to do a high resolution grayscale scan.

In conjunction with the fifth exemplary embodiment, the tokens are stored at very high resolution (600 ppi or greater), and the location at which the tokens are placed in the image is also stored at this resolution. When the image is to be rendered at lower resolution, a scale-to-gray transform is done separately on each token. As described above in the description of the fifth embodiment, the gray anti-aliasing pixels form a boundary that is one (rather than two) pixels wide, and the edges thus appear sharper. The compressed tokens require approximately twice the storage for the image parts, but the location information for the tokens is not substantially increased. Thus, the increase in storage when using tokens at 600 ppi instead of 300 ppi is much less than a factor of two, which was the increase predicted in the implementation of the fifth exemplary embodiment where the very high resolution binary text or lineart mask is compressed as a full image.

The exemplary embodiments of the present invention may also be implemented to generate anti-aliased text and line graphics in color images. In a typical MRC format, both the foreground and background are represented by low-resolution color images.

The first exemplary embodiment uses the actual scanned values of the boundary pixels. This provides the best results for the rendered boundary pixels, but from a storage view, it is the most storage-wasteful method. When tokens are used, only one set of boundary pixels is stored for each token. For grayscale images, this is straightforward, as described above. However, for color, the fact that each token *instance* has a different set of colors in both the foreground and background and requires adaptation of the method for generating anti-aliased text data.

A simple method for color anti-aliased tokens in accordance with the first exemplary embodiment can be used if a color scan at high resolution, e.g., 300 ppi, is available. Therefore, implementation of the first exemplary embodiment may require that

foreground and background images, which are typically scanned or saved at low resolution, have been scaled appropriately so that corresponding pixels for each location can be identified. For each high resolution color boundary pixel on a token, the three colors are stored as three gray values. Each gray value is computed as a fraction of the gray, i.e., intensity between low-frequency foreground and background images at that location. For example, if a red intensity is halfway between foreground and background red intensities, the red gray value is taken to be 0.5. It also foreseeable that the gray value can be stored with quantized values between 0 and 2 - 1, for some value of n. These three gray values are thus used for interpolation between foreground and background pixels on the boundary pixels of each instance of the token.

Assume that the foreground and background pixel saturation values for red are S^r_{fore} and S^r_{back}, and similarly for blue and green. Also assume the boundary pixel saturation value for red is S^r_{bdry}. Then the fraction $S^r_{bdry}/(S^r_{fore} - S^r_{back})$ is stored and the red component of this boundary pixel in each token instance is rendered using this fraction and the local values of S^r_{fore} and S^r_{back}.

However, it may not be practical to scan the image with high resolution color so that these three color values are not available separately for each boundary pixel. Therefore, assume instead that only a high resolution grayscale image is available. Each boundary pixel is a weighted average of the contributions from the three, unavailable color values. In such a situation, this grayscale value can be used to store a single value for each token pixel. Specifically, the weighting factors appropriate to the scanner are used to find the weighted average saturations of the nearest pixels in the foreground and background color images. These saturation values are then compared with the high resolution grayscale value for the boundary pixel. Subsequently, for each boundary pixel on the token, an interpolation factor between foreground and background is stored. However, in this implementation, it is a single number. For rendering, this interpolation factor is selected for the boundary pixel of each token instance, and is used to find the red, green and blue intensities by interpolating between the foreground and background low-resolution color images.

The crucial element is that the boundary pixels of the token are labeled with parameter values that allow an approximate reconstruction of the actual color for each token instance, which has different foreground and background colors. A linear interpolation, as described above, is simple, but other weighted interpolation methods could also be used.

Although the application of the first exemplary embodiment to color images has been presented, one of ordinary skill in the art would readily recognize how the second through fifth embodiments may be implemented to provide color anti-aliased tokens. The basic idea when using tokens is to save just the gray value, or if appropriate, three gray values, and to use this information to determine how similar the rendered boundary pixel is to the foreground and background pixels in that vicinity.

When using the full image mask instead of tokens, the first exemplary embodiment can be implemented in two different ways. The straightforward method is to save and use the color boundary pixels directly. However, if only the grayscale value is available, the grayscale data can be used to interpolate between foreground and background, as described above for tokens. When implementing the second through fifth embodiments in conjunction with the full image mask, interpolation must be performed using the computed gray value, since the actual color or grayscale values of the high resolution boundary pixels are not available.

The exemplary embodiments of the present invention provide methods and systems for adding anti-aliasing to MRC text or lineart masks. These methods improve the appearance of a resulting MRC image, both on display, at high resolution and under typical viewing conditions which are sub-sampled, and on printed output. Although the experimental result examples shown in some of the Figures indicate potential display output for the various exemplary embodiments, the ability to capture on paper the quality of the visual image as it would appear on a 80 ppi display is limited. For example, a binary 300 ppi printer was used to render the images and the rendering algorithm PostScript™ applied halftone filters to render the gray pixels. Each gray pixel on the display is rendered approximately by a 4x4 halftoned region.

Additionally, although methods of image rendering have been explained above to indicate that compressed data is stored to memory 540 and subsequently decoded and used to render images, it is not essential that the compressed data be stored in memory 540. Rather, it is foreseeable that the compressed data provided by the encoder 530 may be provided directly to the decoder 530 to reduce the amount of data to be transmitted to the decoder, e.g., when the decoder is actually part of an image renderer. Such an application would be beneficial when data space in the connection to the image render is limited.

WHAT IS CLAIMED IS:

1. An image rendering system comprising:

a scanner that scans an image and produces image data;

an encoder that is coupled to the scanner and encodes the image data to provide encoded image data including anti-aliased grayscale text or lineart data that includes an identification of boundary pixels and associated pixel values; and

a decoder that is coupled to the encoder and decodes the encoded image data to provide decompressed data including anti-aliased text or lineart data and renders the decompressed data.

2. The image rendering system of claim 1, wherein the system implements an MRC image architecture.

3. The image rendering system of claim 1, further comprising memory that is coupled to the encoder and decoder and that stores the encoded image data, the memory being coupled to the decoder.

4. The image rendering system of claim 1, wherein the scanner scans an image and produces high resolution grayscale data, and wherein the encoder separates the grayscale pixels of the high resolution grayscale data into boundary pixels and non-boundary pixels, individually derives values of grayscale boundary pixels using the high resolution grayscale data, and stores the individually derived values of the grayscale boundary pixels.

5. The image rendering system of claim 4, wherein the encoder compresses the individually derived values of the grayscale boundary pixels as part of a set of grayscale tokens that represent connected components in a foreground image that is part of the scanned image data.

6. The image rendering system of claim 5, wherein the image data is color image data and a foreground image and a background image of the scanned image data are represented by low-resolution color data.

7. The image rendering system of claim 5, wherein, during compression, the encoder selects the boundary pixels from the scanned image data, quantizes the boundary pixels and analyzes each token as both a foreground layer and a boundary layer.

8. The image rendering system of claim 7, wherein the foreground layer and the boundary layer are compressed separately.

9. The image rendering system of claim 5, wherein only one set of boundary pixels is stored for each token.

10. The image rendering system of claim 6, wherein, for each high resolution color boundary pixel on a token, three colors are stored as three grayscale values.

11. The image rendering system of claim 10, wherein each grayscale value is computed as a fraction of an intensity between the foreground image and the background image.

12. The image rendering system of claim 10, wherein grayscale values of the boundary pixels are used to interpolate between foreground and background images during rendering of the image data.

13. The image rendering system of claim 12, wherein the three grayscale values are used for interpolation between foreground and background pixels on the boundary pixels of each instance of the token.

14. The image rendering system of claim 4, wherein the step of storing the individually derived values of the grayscale boundary pixels includes storing a full image mask corresponding to the scanned image data.

15. The image rendering system of claim 1, wherein the scanner scans the image and produces high resolution grayscale data and the encoder separates the grayscale pixels of the high resolution grayscale data into boundary pixels and non-boundary pixels, separates the grayscale boundary pixels into interior boundary pixels and exterior boundary pixels, determines boundary pixel connectedness for both the interior boundary pixels and the exterior boundary pixels, determines a representative grayscale value for the interior boundary pixels and a representative grayscale value for the exterior boundary pixels based on the connectedness of the interior boundary pixels and the exterior boundary pixels respectively and stores the grayscale values for the boundary pixels.

16. The image rendering system of claim 15, wherein the encoder stores the individually derived values of the grayscale boundary pixels by compressing the individually derived values of the grayscale boundary pixels as part of a set of grayscale

tokens that represent connected components in a foreground image that is part of the scanned image data.

17. The image rendering system of claim 1, wherein the scanner scans the image and produces high resolution binary data and the encoder individually estimates median boundary pixel values based on a number of oppositely colored four neighbor pixels to each of the boundary pixels and stores the estimated median boundary pixel values.

18. The image rendering system of claim 17, wherein the encoder stores the estimated median boundary pixel values by compressing the estimated median boundary pixel values as part of a set of grayscale tokens that represent connected components in a foreground image that is part of the scanned image data.

19. The image rendering system of claim 18, wherein the decoder performs rendering of the grayscale tokens by analyzing a connectivity of the boundary pixels and performing grayscale substitutions.

20. The image rendering system of claim 19, wherein substitution values used during performing grayscale substitutions are known a priori.

21. The image rendering system of claim 17, wherein the encoder stores the estimated median boundary pixel values by compressing the estimated median boundary pixel values as part of a foreground mask that represents connected components in a foreground image that is part of the scanned image data.

22. The image rendering system of claim 21, wherein the decoder performs rendering of the foreground mask by analyzing a connectivity of the boundary pixels and performing grayscale substitutions.

23. The image rendering system of claim 1, wherein the scanner scans the image and produces high resolution binary data and the encoder separates the boundary pixels into interior boundary pixels and exterior boundary pixels.

24. The image rendering system of claim 23, wherein the encoder determines a first global grayscale value corresponding to the interior boundary pixels and a second global grayscale value corresponding to the exterior boundary pixels and stores the interior and exterior boundary pixel data including the first and second grayscale boundary pixel values.

25. The image rendering system of claim 23, wherein the decoder stores default global grayscale values including first and second default global grayscale values and sets the interior boundary pixels to the first default global grayscale value and sets the exterior boundary pixels to the second global default grayscale value.

26. The image rendering system of claim 24, wherein the first and second default global values are determined based on analysis of image data other than the scanned image data.

27. The image rendering system of claim 23, wherein the decoder renders the image using the interior and exterior boundary pixel values and the binary high resolution data.

28. The image rendering system of claim 1, wherein the scanner scans the image and produces very high resolution binary data and the encoder converts the very high resolution binary data to high resolution grayscale data and stores the high resolution grayscale data.

29. The image rendering system of claim 28, wherein the encoder stores the high resolution grayscale data by compressing the high resolution grayscale data as part of a set of grayscale tokens that represent connected components in a foreground image that is part of the scanned image data.

30. An image rendering method comprising:

scanning an image to obtain scanned image data including text or lineart data;

generating an anti-aliased grayscale version of the text or lineart data including determining pixel values of the boundary pixels in the anti-aliased grayscale version of the scanned text or lineart data;

rendering the image using the determined pixel values.

31. The image rendering method of claim 30, wherein the step of scanning the image produces high resolution grayscale data and generating an anti-aliased grayscale version of the text or lineart data comprises:

separating the grayscale pixels of the high resolution grayscale data into boundary pixels and non-boundary pixels;

individually deriving values of grayscale boundary pixels using the high resolution grayscale data; and

storing the individually derived values of the grayscale boundary pixels.

32. The image rendering method of claim 31, wherein the step of storing the individually derived values of the grayscale boundary pixels includes compressing the individually derived values of the grayscale boundary pixels as part of a set of grayscale tokens that represent connected components in a foreground image that is part of the scanned image data.

33. The image rendering method of claim 32, wherein at least one of a token index, location, color, character and font ID and anti-aliasing information are stored with each token instance.

34. The image rendering method of claim 32, wherein the image data is color image data and a foreground image and a background image of the scanned image data are represented by low-resolution color data.

35. The image rendering method of claim 34, wherein, for each high resolution color boundary pixel on a token, three colors are stored as three grayscale values.

36. The image rendering method of claim 35, wherein each grayscale value is computed as a fraction of an intensity between the foreground image and the background image.

37. The image rendering method of claim 35, wherein grayscale values of the boundary pixels are used to interpolate between foreground and background images during rendering of the image data.

38. The image rendering method of claim 35, wherein the three grayscale values are used for interpolation between foreground and background pixels on the boundary pixels of each instance of the token.

39. The image rendering method of claim 31, wherein the step of storing the individually derived values of the grayscale boundary pixels includes storing a full image mask corresponding to the scanned image data.

40. The image rendering method of claim 30, wherein the step of scanning the image produces high resolution grayscale data and the step generating an anti-aliased grayscale version of the text or lineart data comprises:

separating the grayscale pixels of the high resolution grayscale data into boundary pixels and non-boundary pixels;

separating the grayscale boundary pixels into interior boundary pixels and exterior boundary pixels;

determining grayscale boundary pixel connectedness by separately analyzing the interior grayscale boundary pixels and the exterior grayscale boundary pixels;

individually deriving values of grayscale boundary pixels using the high resolution grayscale data and the determined grayscale boundary pixel connectedness; and

storing the individually derived values of the grayscale boundary pixels.

41. The image rendering method of claim 40, wherein the step of storing the individually derived values of the grayscale boundary pixels includes compressing the individually derived values of the grayscale boundary pixels as part of a set of grayscale tokens that represent connected components in a foreground image that is part of the scanned image data.

42. The image rendering method of claim 41, wherein at least one of a token index, location, color, character and font ID and anti-aliasing information are stored with each token instance.

43. The image rendering method of claim 41, wherein the step of rendering is performed by rendering grayscale tokens by analyzing a connectivity of the boundary pixels and performing grayscale substitutions.

44. The image rendering method of claim 30, wherein the step of scanning the image produces high resolution binary data and the step of generating an anti-aliased grayscale version of the text or lineart data comprises:

individually estimating median boundary pixel values based on a number of oppositely colored four neighbor pixels to each of the boundary pixels; and

storing the estimated median boundary pixel values.

45. The image rendering method of claim 44, wherein the step of storing the estimated median boundary pixel values stores the estimated median boundary pixel values as part of a set of grayscale tokens that represent connected components in a foreground image that is part of the scanned image data.

46. The image rendering method of claim 45, wherein at least one of a token index, location, color, character and font ID and anti-aliasing information are stored with each token instance.

47. The image rendering method of claim 45, wherein the step of rendering is performed by rendering the grayscale tokens by analyzing a connectivity of the boundary pixels and performing grayscale substitutions.

48. The image rendering method of claim 47, wherein substitution values used during performing grayscale substitutions are known a priori.

49. The image rendering method of claim 32, wherein the step of scanning the image produces high resolution binary data and the step of generating an anti-aliased grayscale version of the text or lineart data comprises:

separating the boundary pixels into interior boundary pixels and exterior boundary pixels;

determining a first global grayscale value corresponding to the interior boundary pixels and a second global grayscale value corresponding to the exterior boundary pixels; and

storing the interior and exterior boundary pixel data including the first and second grayscale boundary pixel values.

50. The image rendering method of claim 49, wherein the step of storing the interior and exterior boundary pixel data stores the interior and exterior boundary pixel data as part of a set of grayscale tokens that represent connected components in a foreground image that is part of the scanned image data.

51. The image rendering method of claim 50, wherein at least one of a token index, location, color, character and font ID and anti-aliasing information are stored with each token instance.

52. The image rendering system of claim 49, further comprising storing default global grayscale values including first and second default global grayscale values and setting the interior boundary pixels to the first default global grayscale value and setting the exterior boundary pixels to the second default global grayscale value.

53. The image rendering system of claim 52, wherein the first and second default global values are determined based on analysis of image data other than the scanned image data.

54. The image rendering method of claim 49, further comprising rendering the image using the interior and exterior boundary pixel values and the binary high resolution data.

55. The image rendering method of claim 30, wherein the step of scanning the image produces very high resolution binary data and the step of generating an anti-aliased grayscale version of the text or lineart data comprises:

converting the very high resolution binary data to high resolution grayscale data; and

storing the high resolution grayscale data.

56. The image rendering method of claim 55, wherein the step of storing the high resolution grayscale data includes compressing the high resolution grayscale data as part of a set of grayscale tokens that represent connected components in a foreground image that is part of the scanned image data.

57. The image rendering method of claim 56, wherein at least one of a token index, location, color, character and font ID and anti-aliasing information are stored with each token instance.

58. The image rendering method of claim 55, wherein the converting step is performed by filtering and down sampling the very high resolution binary text or lineart data to produce high resolution grayscale text or lineart data.

59. The image rendering method of claim 58, wherein the step of filtering and down sampling comprises:

tiling the very high resolution binary text or lineart data prior to subsampling the very high resolution binary text or lineart data, and calculating a gray value for each tile that is proportional to the number of pixels of a first value in the tile.

ABSTRACT

A method and system for storing and generating anti-aliased text and lineart data from compressed document image files, using a MRC model that represents the image as an ordered set of mask/image pairs at resolutions appropriate to the content of each layer. The method and system provide the ability to generate for anti-aliased text data to improve appearance at both high and low resolution, and to avoid baseline jitter of compressed tokens.

D6633 D6633 D6633

FIGURE 1

FIGURE 2

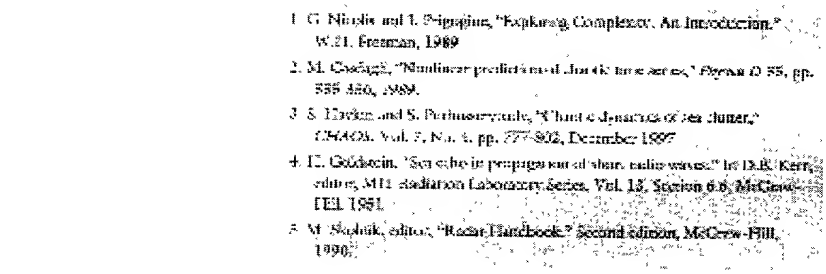
1. G. Nicolis and I. Prigogine, "Exploring Complexity. An Introduction," W.H. Freeman, 1989.
2. M. Casdagli, "Nonlinear prediction of chaotic time series," *Physica D* 35, pp. 335-356, 1989.
3. S. Haykin and S. Puthusserypady, "Chaotic dynamics of sea clutter," *CHAOS*, Vol. 7, No. 4, pp. 777-802, December 1997.
4. H. Goldstein, "Sea echo in propagation of short radio waves," in D.K. Kerr, editor, MIT Radiation Laboratory Series, Vol. 13, Section 6.6, McGraw-Hill, NY, 1951.
5. M. Skolnik, editor, "Radar Handbook," Second edition, McGraw-Hill, 1990.

FIGURE 3

1. G. Nicolis and I. Prigogine, "Exploring Complexity. An Introduction," W.H. Freeman, 1989.
2. M. Casdagli, "Nonlinear prediction of chaotic time series," *Physica D* 35, pp. 335-356, 1989.
3. S. Haykin and S. Puthusserypady, "Chaotic dynamics of sea clutter," *CHAOS*, Vol. 7, No. 4, pp. 777-802, December 1997.
4. H. Goldstein, "Sea echo in propagation of short radio waves," in D.K. Kerr, editor, MIT Radiation Laboratory Series, Vol. 13, Section 6.6, McGraw-Hill, NY, 1951.
5. M. Skolnik, editor, "Radar Handbook," Second edition, McGraw-Hill, 1990.

FIGURE 4

06170 "2852860

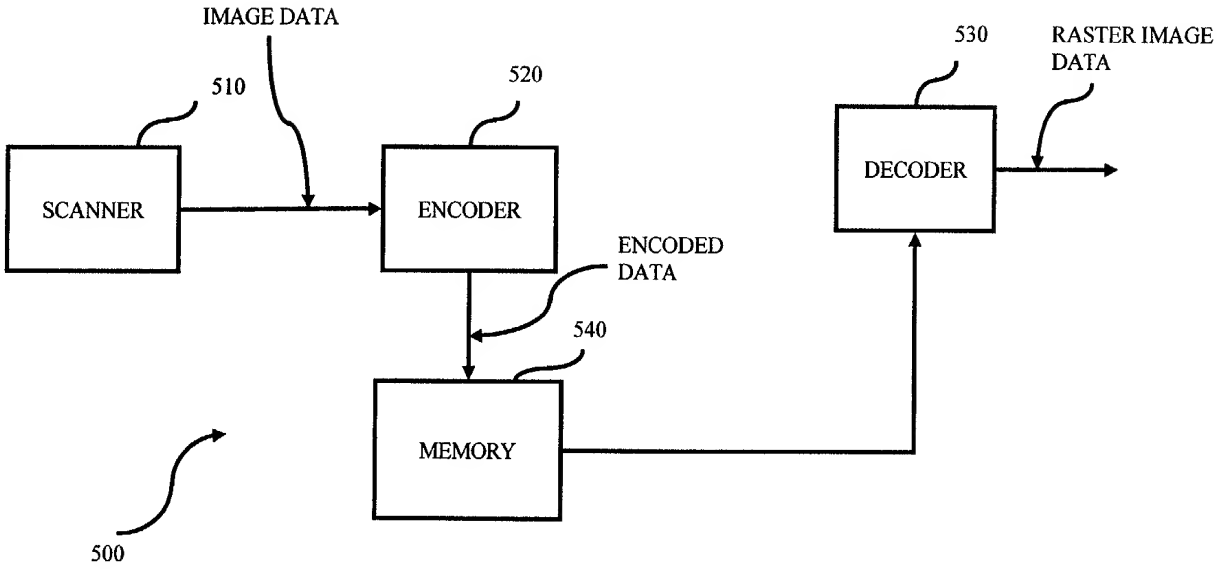

FIGURE 5

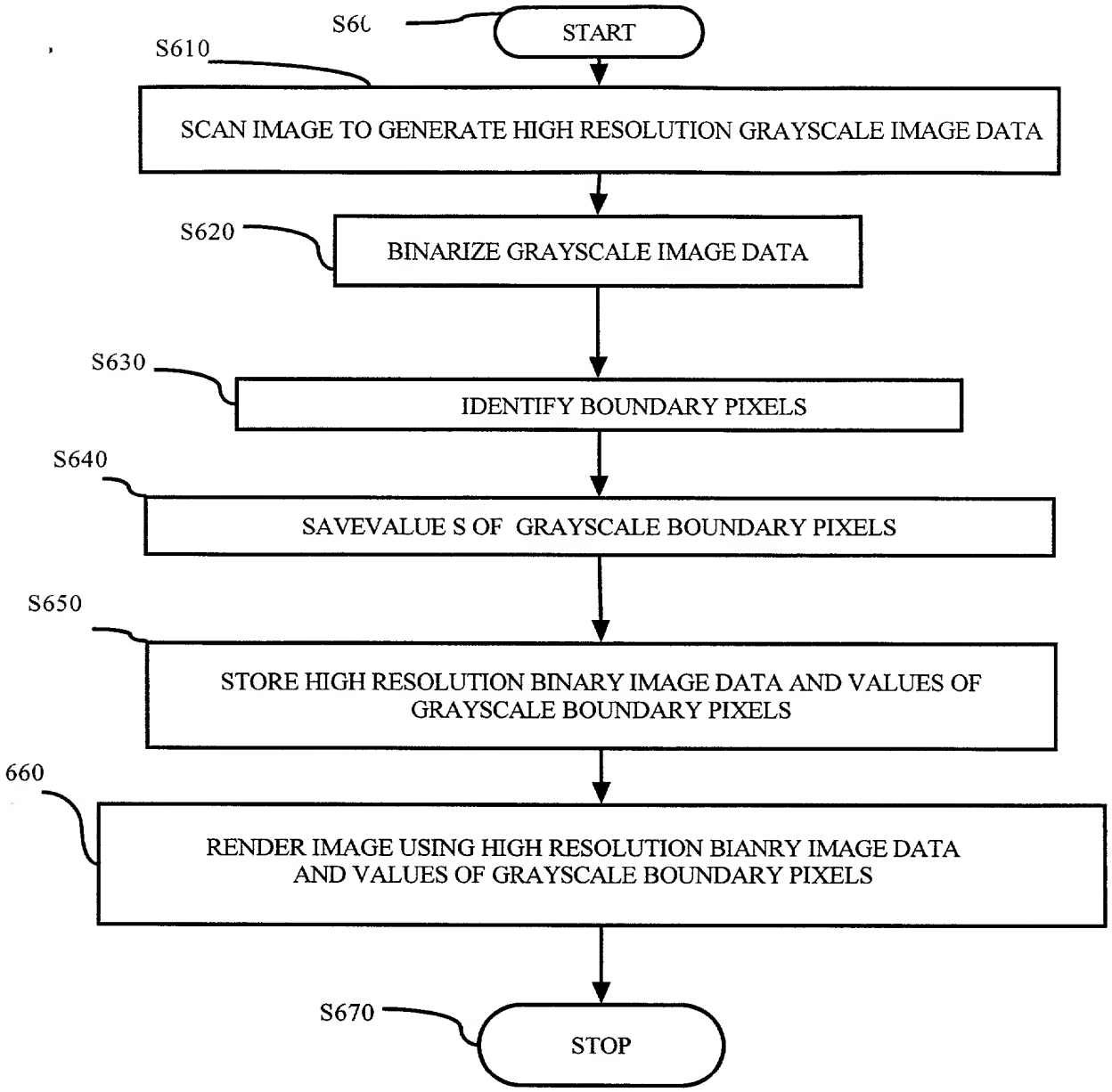
S60
START
S610
SCAN IMAGE TO GENERATE HIGH RESOLUTION GRAYSCALE IMAGE DATA
S620
BINARIZE GRAYSCALE IMAGE DATA
S630
IDENTIFY BOUNDARY PIXELS
S640
SAVEVALUE S OF GRAYSCALE BOUNDARY PIXELS
S650
STORE HIGH RESOLUTION BINARY IMAGE DATA AND VALUES OF GRAYSCALE BOUNDARY PIXELS
660
RENDER IMAGE USING HIGH RESOLUTION BIANRY IMAGE DATA AND VALUES OF GRAYSCALE BOUNDARY PIXELS
S670
STOP

Figure 6

Figure 7

Figure 8

Figure 9

Figure 10

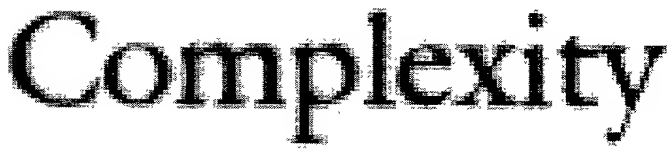

Figure 11

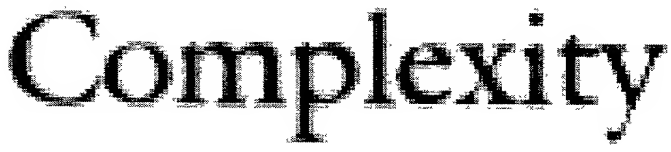

Figure 12

USPTO-3050-0154

Figure 13

1. G. Nicolis and I. Prigogine, "Exploring Complexity. An Introduction," W.H. Freeman, 1989.

2. M. Casdagli, "Nonlinear prediction of chaotic time series," *Physica D* 35, pp. 335-356, 1989.

3. S. Haykin and S. Puthusserypady, "Chaotic dynamics of sea clutter," CHAOS, Vol. 7, No. 4, pp. 777-802, December 1997.

4. H. Goldstein, "Sea echo in propagation of short radio waves." In D.E. Kerr, editor, MIT Radiation Laboratory Series, Vol. 13, Section 6.6, McGraw-Hill, 1951.

5. M. Skolnik, editor, "Radar Handbook," Second edition, McGraw-Hill, 1990.

Figure 14

1. G. Nicolis and I. Prigogine, "Exploring Complexity. An Introduction," W.H. Freeman, 1989.

2. M. Casdagli, "Nonlinear prediction of chaotic time series," *Physica D* 35, pp. 335-356, 1989.

3. S. Haykin and S. Puthusserypady, "Chaotic dynamics of sea clutter," CHAOS, Vol. 7, No. 4, pp. 777-802, December 1997.

4. H. Goldstein, "Sea echo in propagation of short radio waves." In D.E. Kerr, editor, MIT Radiation Laboratory Series, Vol. 13, Section 6.6, McGraw-Hill, 1951.

5. M. Skolnik, editor, "Radar Handbook," Second edition, McGraw-Hill, 1990.

Figure 15

1. G. Nicolis and I. Prigogine, "Exploring Complexity. An Introduction," W.H. Freeman, 1989.

2. M. Casdagli, "Nonlinear prediction of chaotic time series," *Physica D* 35, pp. 335-356, 1989.

3. S. Haykin and S. Puthusserypady, "Chaotic dynamics of sea clutter," CHAOS, Vol. 7, No. 4, pp. 777-802, December 1997.

4. H. Goldstein, "Sea echo in propagation of short radio waves." In D.E. Kerr, editor, MIT Radiation Laboratory Series, Vol. 13, Section 6.6, McGraw-Hill, 1951.

5. M. Skolnik, editor, "Radar Handbook," Second edition, McGraw-Hill, 1990.

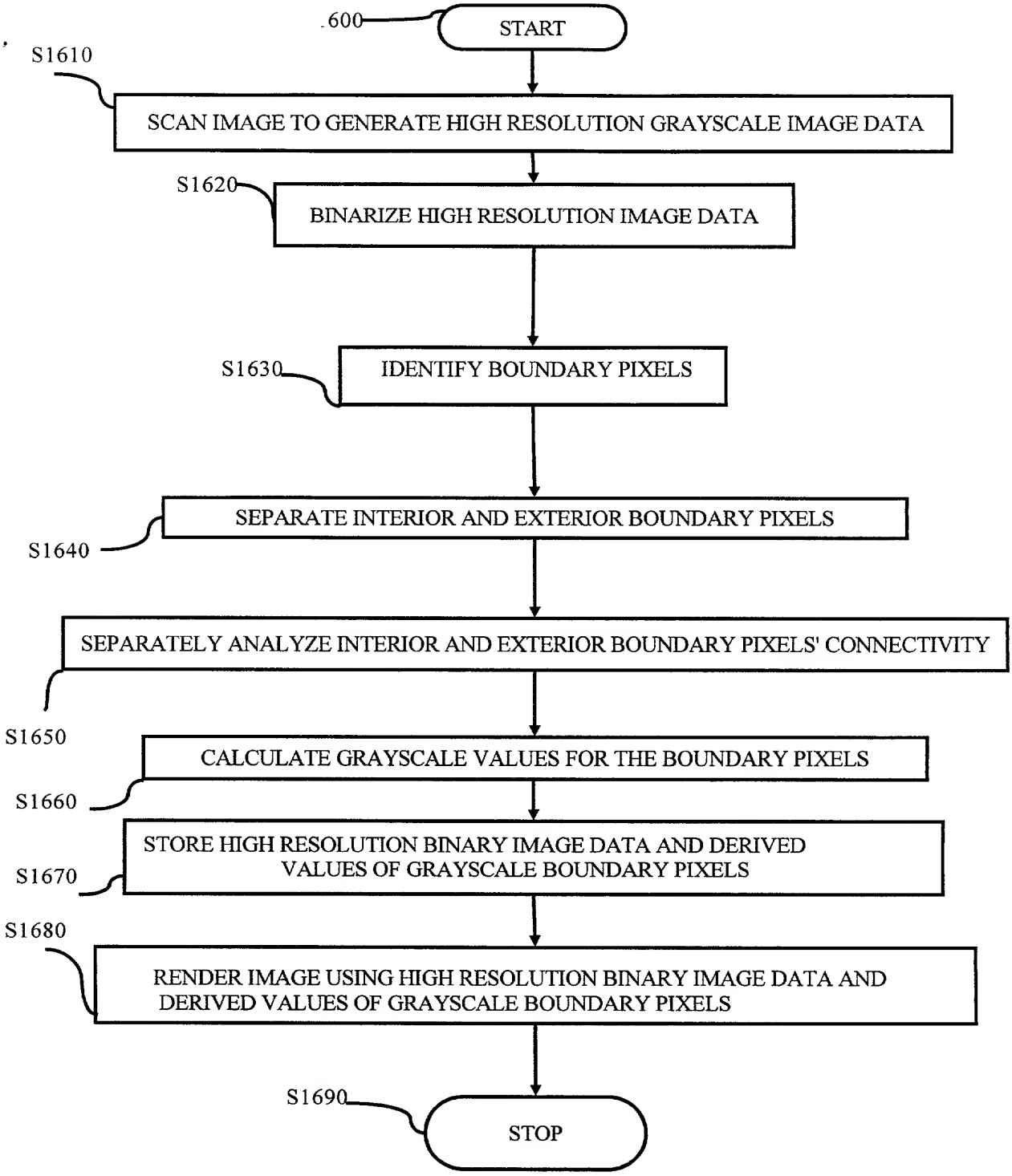
600
START
S1610
SCAN IMAGE TO GENERATE HIGH RESOLUTION GRAYSCALE IMAGE DATA
S1620
BINARIZE HIGH RESOLUTION IMAGE DATA
S1630
IDENTIFY BOUNDARY PIXELS
SEPARATE INTERIOR AND EXTERIOR BOUNDARY PIXELS
S1640
SEPARATELY ANALYZE INTERIOR AND EXTERIOR BOUNDARY PIXELS' CONNECTIVITY
S1650
CALCULATE GRAYSCALE VALUES FOR THE BOUNDARY PIXELS
S1660
STORE HIGH RESOLUTION BINARY IMAGE DATA AND DERIVED VALUES OF GRAYSCALE BOUNDARY PIXELS
S1670
S1680
RENDER IMAGE USING HIGH RESOLUTION BINARY IMAGE DATA AND DERIVED VALUES OF GRAYSCALE BOUNDARY PIXELS
S1690
STOP

Figure 16

0677 0: 6852 8h60

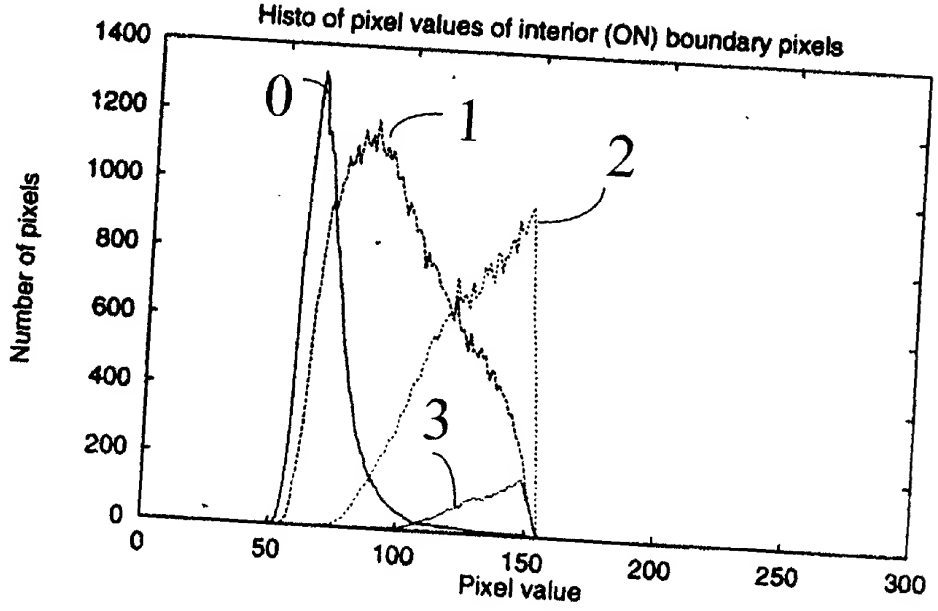

Figure 17

006707 E85Z8h60

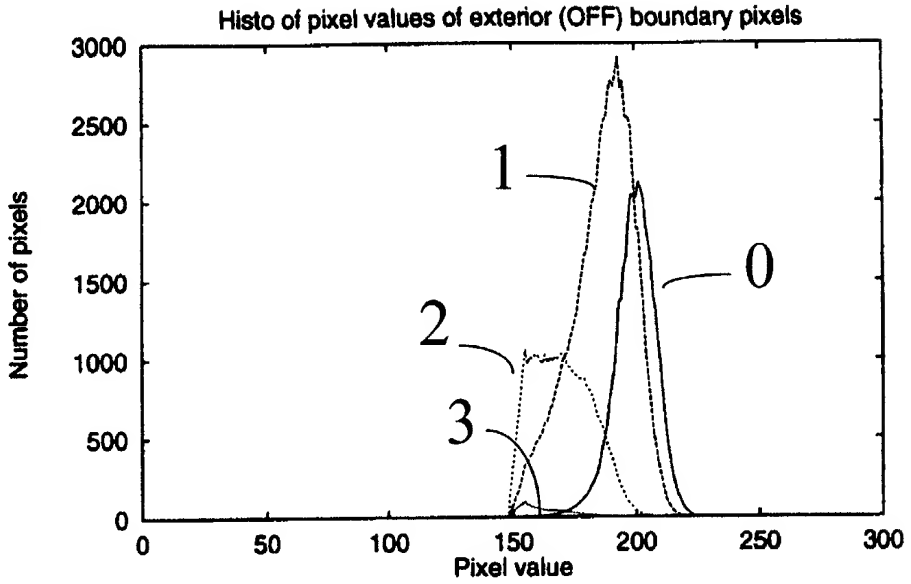

Figure 18

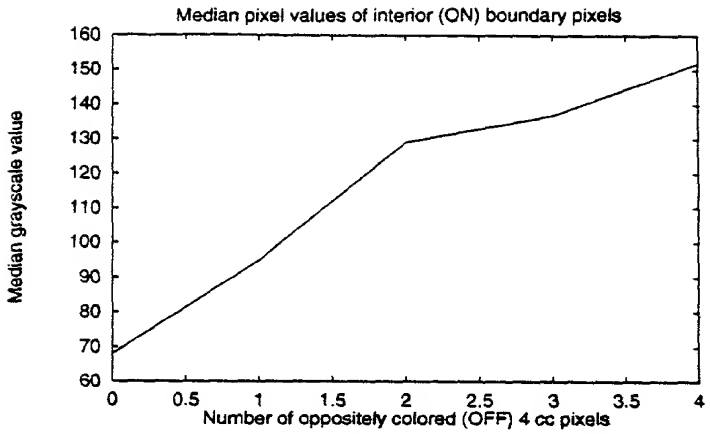
Median pixel values of interior (ON) boundary pixels
Median grayscale value
160
150
140
130
120
110
100
90
80
70
60
0
0.5
1
1.5
2
2.5
3
3.5
4
Number of oppositely colored (OFF) 4 cc pixels

Figure 19

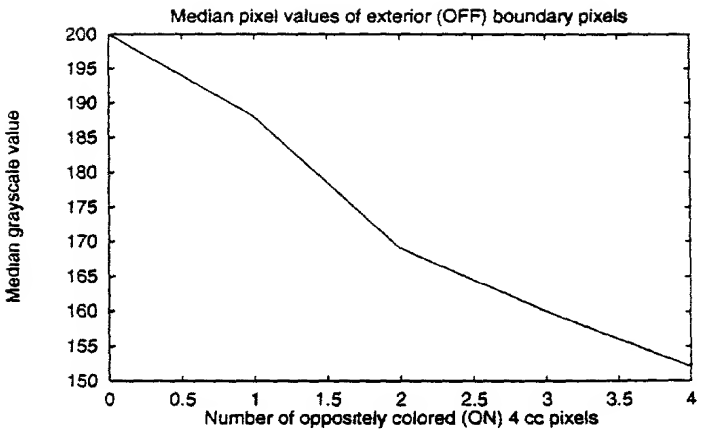
Median pixel values of exterior (OFF) boundary pixels
Median grayscale value
200
195
190
185
180
175
170
165
160
155
150
0
0.5
1
1.5
2
2.5
3
3.5
4
Number of oppositely colored (ON) 4 cc pixels

Figure 20

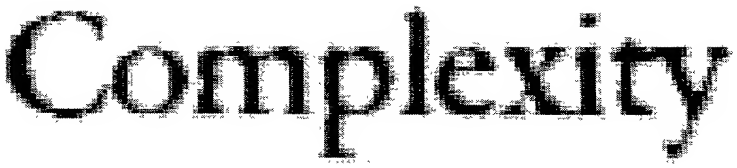

FIGURE 21

1. G. Nicolis and I. Prigogine, "Exploring Complexity: An Introduction," W.H. Freeman, 1989.

2. M. Casdagli, "Nonlinear prediction of chaotic time series," *Physica D* 35, pp. 335-356, 1989.

3. S. Haykin and S. Puthusserypady, "Chaotic dynamics of sea clutter," *CHAOS*, Vol. 7, No. 4, pp. 777-802, December 1997.

4. H. Goldstein, "Sea echo in propagation of short radio waves," In D.R. Kerr, editor, MIT Radiation Laboratory Series, Vol. 13, Section 6.6, McGraw-Hill, 1951.

5. M. Skolnik, editor, "Radar Handbook," Second edition, McGraw-Hill, 1990.

FIGURE 22

FIGURE 23

1. G. Nicolis and I. Prigogine, "Exploring Complexity: An Introduction," W.H. Freeman, 1989.

2. M. Casdagli, "Nonlinear prediction of chaotic time series," *Physica D* 35, pp. 335-356, 1989.

3. S. Haykin and S. Puthusserypady, "Chaotic dynamics of sea clutter," *CHAOS*, Vol. 7, No. 4, pp. 777-802, December 1997.

4. H. Goldstein, "Sea echo in propagation of short radio waves," In D.R. Kerr, editor, MIT Radiation Laboratory Series, Vol. 13, Section 6.6, McGraw-Hill, 1951.

5. M. Skolnik, editor, "Radar Handbook," Second edition, McGraw-Hill, 1990.

FIGURE 24

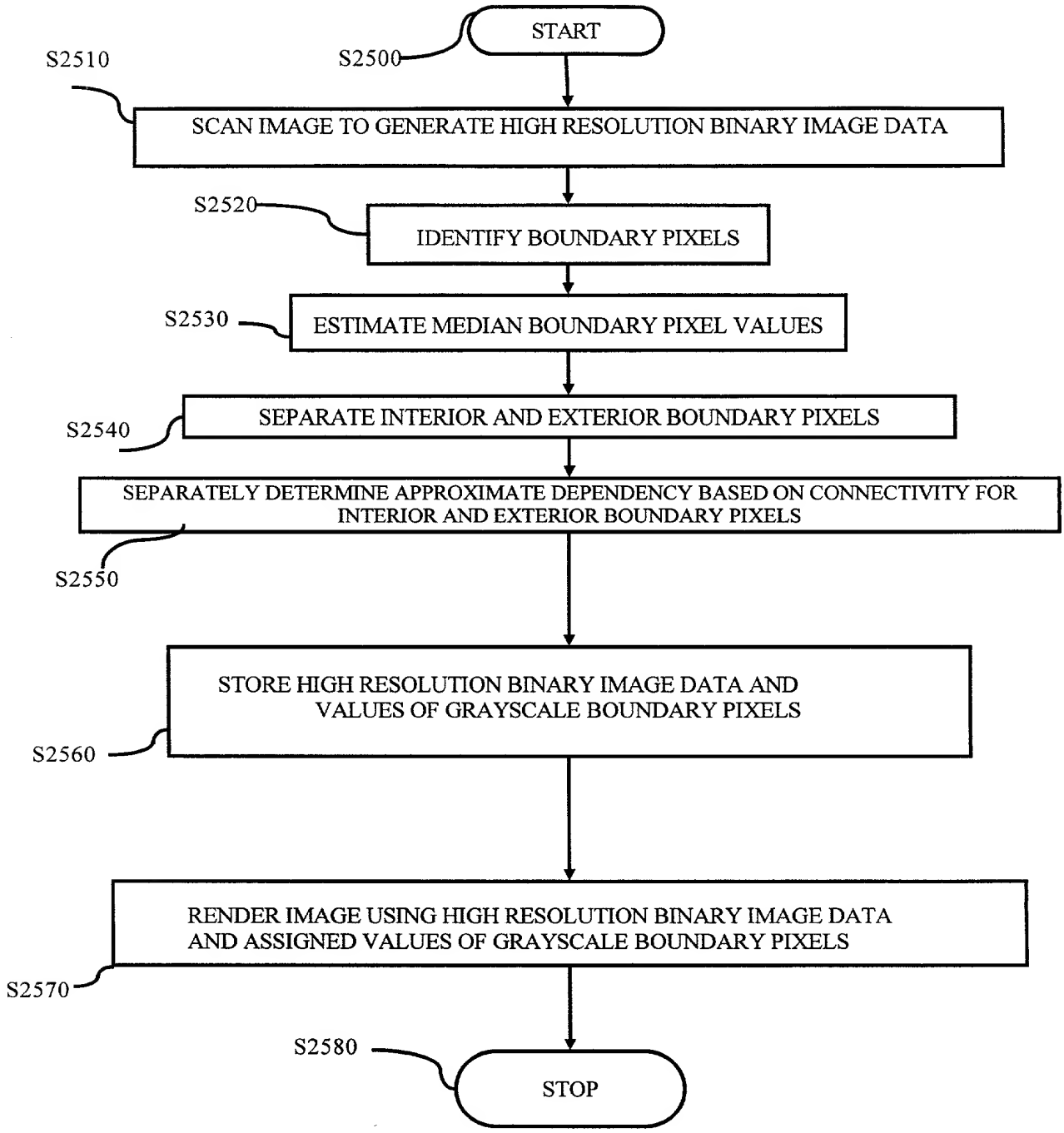
START
S2500
S2510
SCAN IMAGE TO GENERATE HIGH RESOLUTION BINARY IMAGE DATA
S2520
IDENTIFY BOUNDARY PIXELS
S2530
ESTIMATE MEDIAN BOUNDARY PIXEL VALUES
S2540
SEPARATE INTERIOR AND EXTERIOR BOUNDARY PIXELS
SEPARATELY DETERMINE APPROXIMATE DEPENDENCY BASED ON CONNECTIVITY FOR INTERIOR AND EXTERIOR BOUNDARY PIXELS
S2550
STORE HIGH RESOLUTION BINARY IMAGE DATA AND VALUES OF GRAYSCALE BOUNDARY PIXELS
S2560
RENDER IMAGE USING HIGH RESOLUTION BINARY IMAGE DATA AND ASSIGNED VALUES OF GRAYSCALE BOUNDARY PIXELS
S2570
S2580
STOP

Figure 25

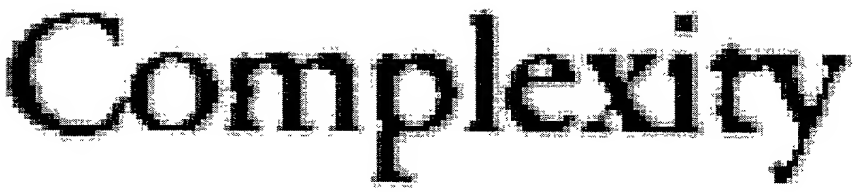

FIGURE 26

1. G. Nicolis and I. Prigogine, "Exploring Complexity: An Introduction," W.H. Freeman, 1989.

2. M. Casdagli, "Nonlinear prediction of chaotic time series," *Physica D* 35, pp. 335-356, 1989.

3. S. Haykin and S. Puthusserypady, "Chaotic dynamics of sea clutter," *CHAOS*, Vol. 7, No. 4, pp. 777-802, December 1997.

4. H. Goldstein, "Sea echo in propagation of short radio waves," in D.E. Kerr, editor, MIT Radiation Laboratory Series, Vol. 13, Section 6.6, McGraw-Hill, 1951.

5. M. Skolnik, editor, "Radar Handbook," Second edition, McGraw-Hill, 1990.

FIGURE 27

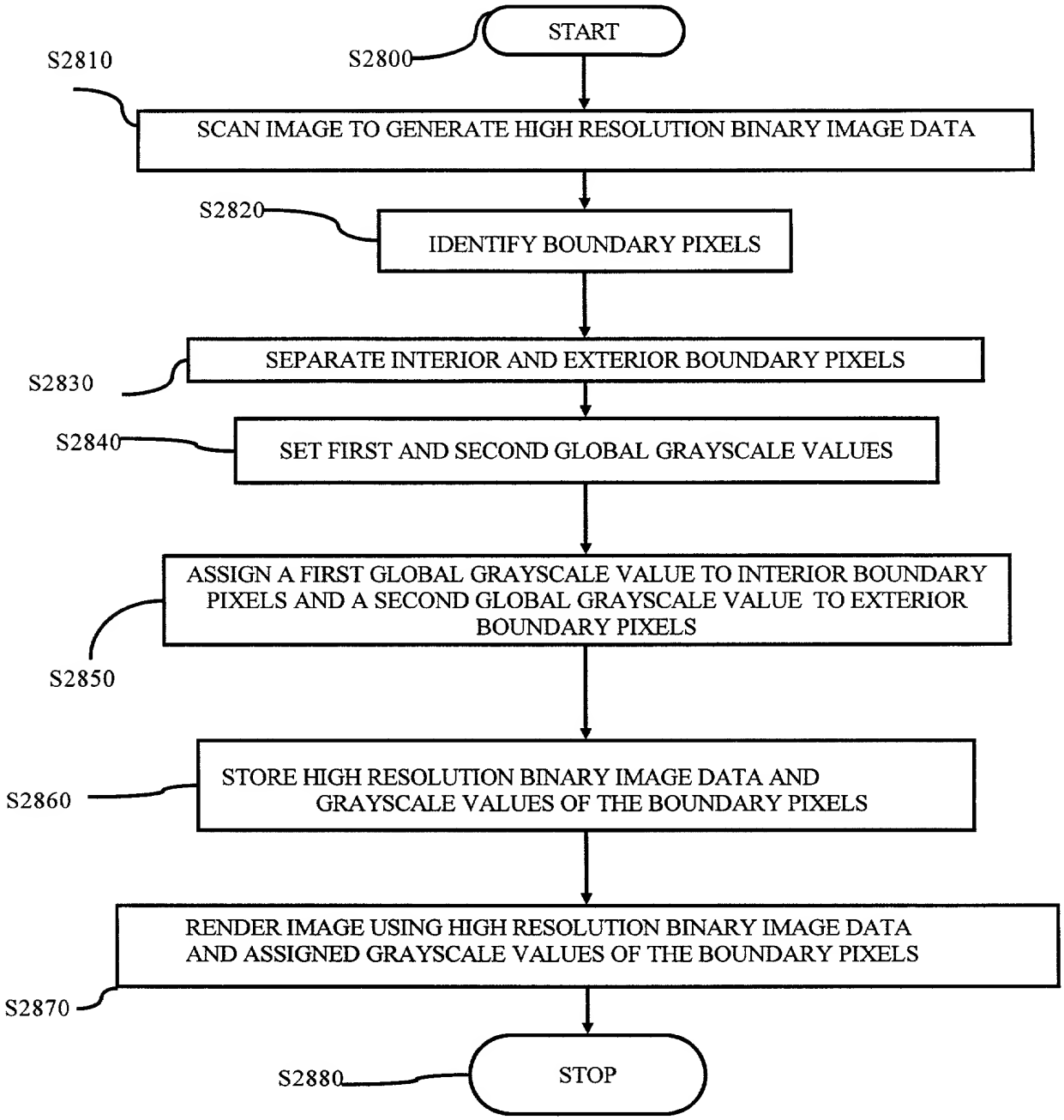
START
S2800
S2810
SCAN IMAGE TO GENERATE HIGH RESOLUTION BINARY IMAGE DATA
S2820
IDENTIFY BOUNDARY PIXELS
SEPARATE INTERIOR AND EXTERIOR BOUNDARY PIXELS
S2830
S2840
SET FIRST AND SECOND GLOBAL GRAYSCALE VALUES
ASSIGN A FIRST GLOBAL GRAYSCALE VALUE TO INTERIOR BOUNDARY PIXELS AND A SECOND GLOBAL GRAYSCALE VALUE TO EXTERIOR BOUNDARY PIXELS
S2850
S2860
STORE HIGH RESOLUTION BINARY IMAGE DATA AND GRAYSCALE VALUES OF THE BOUNDARY PIXELS
RENDER IMAGE USING HIGH RESOLUTION BINARY IMAGE DATA AND ASSIGNED GRAYSCALE VALUES OF THE BOUNDARY PIXELS
S2870
S2880
STOP

Figure 28

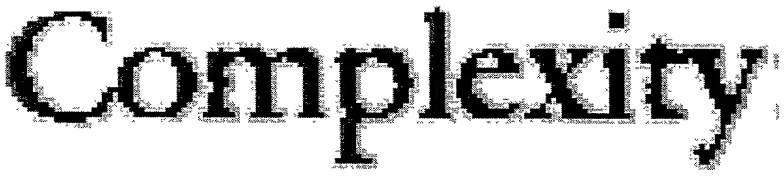

FIGURE 29

1. G. Nicolis and I. Prigogine, "Exploring Complexity: An Introduction," W.H. Freeman, 1989.

2. M. Casdagli, "Nonlinear prediction of chaotic time series," *Physica D* 35, pp. 335-356, 1989.

3. S. Haykin and S. Puthusserypady, "Chaotic dynamics of sea clutter," *CHAOS*, Vol. 7, No. 4, pp. 777-802, December 1997.

4. E. Goldstein, "Sea echo in propagation of short radio waves," In D.E. Kerr, editor, MIT Radiation Laboratory Series, Vol. 13, Section 6.6, McGraw-Hill, 1951.

5. M. Skolnik, editor, "Radar Handbook," Second edition, McGraw-Hill, 1990.

FIGURE 30

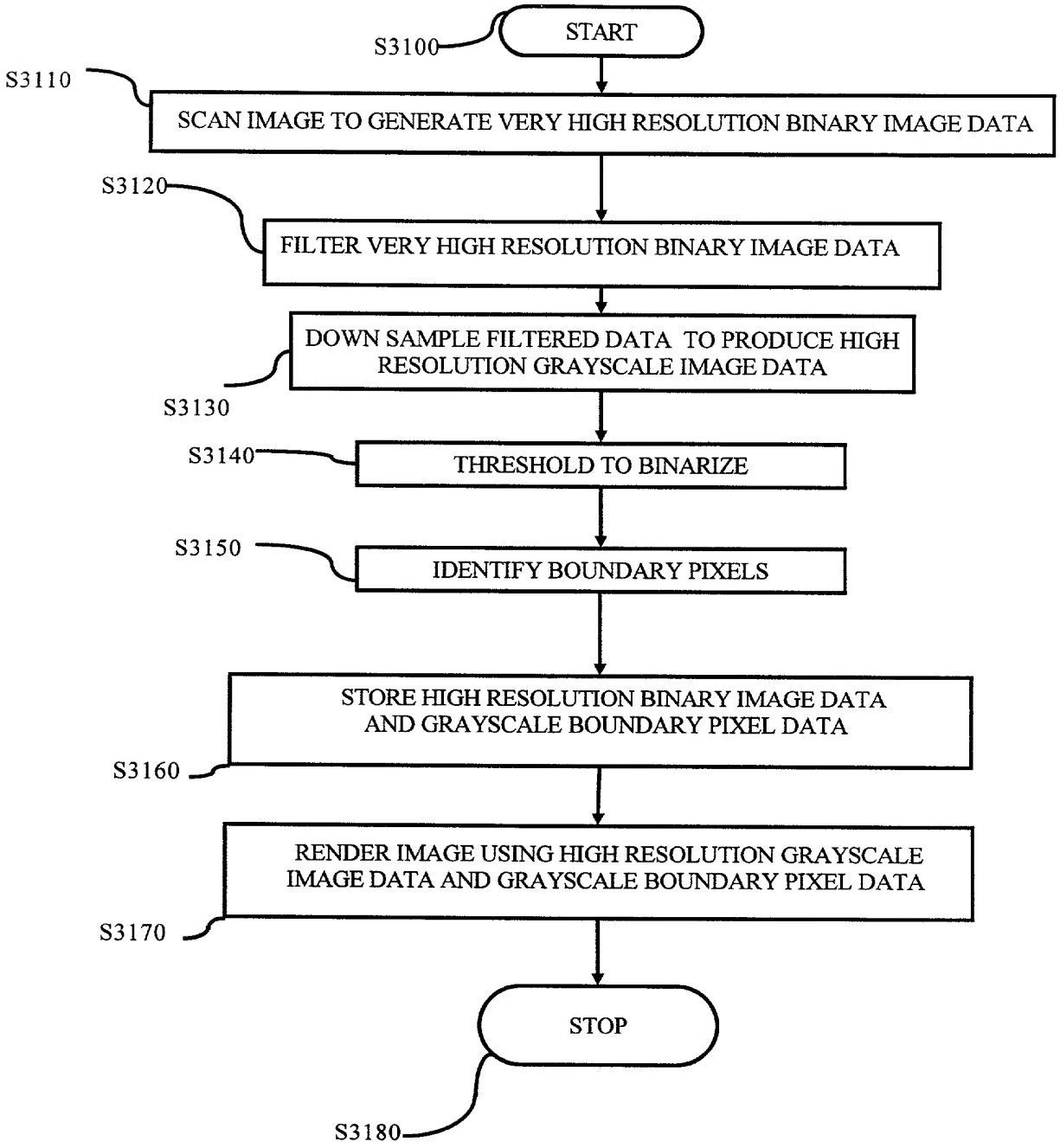
START
S3100
S3110
SCAN IMAGE TO GENERATE VERY HIGH RESOLUTION BINARY IMAGE DATA
S3120
FILTER VERY HIGH RESOLUTION BINARY IMAGE DATA
DOWN SAMPLE FILTERED DATA TO PRODUCE HIGH RESOLUTION GRAYSCALE IMAGE DATA
S3130
S3140
THRESHOLD TO BINARIZE
S3150
IDENTIFY BOUNDARY PIXELS
STORE HIGH RESOLUTION BINARY IMAGE DATA AND GRAYSCALE BOUNDARY PIXEL DATA
S3160
RENDER IMAGE USING HIGH RESOLUTION GRAYSCALE IMAGE DATA AND GRAYSCALE BOUNDARY PIXEL DATA
S3170
STOP
S3180

Figure 31

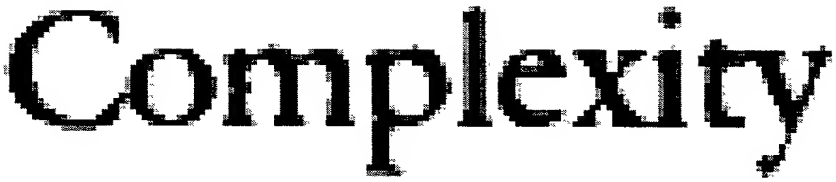

FIGURE 32

1. G. Nicolis and I. Prigogine, "Exploring Complexity. An Introduction," W.H. Freeman, 1989.

2. M. Casdagli, "Nonlinear prediction of chaotic time series," Physica D 35, pp. 335-356, 1989.

3. S. Haykin and S. Puthusserypady, "Chaotic dynamics of sea clutter," CHAOS, Vol. 7, No. 4, pp. 777-802, December 1997.

4. H. Goldstein, "Scattering by propagation of plane radio waves," In D.R. Kerr, editor, MIT Radiation Laboratory Series, Vol. 13, Section 6.6, McGraw-Hill, 1951.

5. M. Skolnik, editor, "Radar Handbook," Second edition, McGraw-Hill, 1990.

FIGURE 33

Docket No.: 104324

APPLICATION FOR UNITED STATES PATENT
DECLARATION AND POWER OF ATTORNEY

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name; that

I verily believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural inventors are named below) of the subject matter which is claimed and for which a patent is sought on the invention entitled:

METHODS FOR GENERATING ANTI-ALIASED TEXT AND LINE GRAPHICS IN COMPRESSED DOCUMENT IMAGES

described and claimed in the specification:

Check one

*a. ☒ attached hereto.

b. ☐ filed on _____ as Application No. _____ and amended on _____ (if applicable).

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose to the Office all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, §1.56. Under Title 35, U.S. Code §119, the priority benefits of the following foreign application(s) and/or United States provisional application(s) filed by me or my legal representatives or assigns within one year prior to this application are hereby claimed:

-NONE-

The following application(s) for patent or inventor's certificate on this invention were filed in countries foreign to the United States of America either (a) more than one year prior to this application, or (b) before the filing date of the above-named foreign priority application(s) and/or United States provisional application(s):

-NONE-

I hereby appoint the following as my attorneys of record with full power of substitution and revocation to prosecute this application and to transact all business in the Patent Office:

James A. Oliff, Reg. No. 27,075; William P. Berridge, Reg. No. 30,024;
Kirk M. Hudson, Reg. No. 27,562; Thomas J. Pardini, Reg. No. 30,411;
Edward P. Walker, Reg. No. 31,450; Robert A. Miller, Registration No. 32,771;
Mario A. Costantino, Registration No. 33,565; Caroline D. Dennison, Registration No. 34,494;
and John Beck, Reg. No. 22,833.

ALL CORRESPONDENCE IN CONNECTION WITH THIS APPLICATION SHOULD BE SENT TO OLIFF & BERRIDGE, PLC, P.O. BOX 19928, ALEXANDRIA, VIRGINIA 22320, TELEPHONE (703) 836-6400.

I hereby declare that I have reviewed and understand the contents of this Declaration, and that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

1 ***Typewritten Full Name of First or Sole Inventor***

Dan	S.	BLOOMBERG
Given Name	Middle Initial	Family Name

2 ****INVENTOR'S SIGNATURE:** Dan S. Bloomberg

3 ****DATE OF SIGNATURE:**

December	30	1999
Month	Day	Year

Residence:	Palo Alto	California	USA
	City	State or Province	Country
Citizenship:	US		

Post Office Address:
(Insert complete mailing address, including country)

1013 Paradise Way
Palo Alto, CA 94306

***This form may be executed only when attached to the specification (including claims) at the end thereof if Box a. is checked.**

****Note to Inventor: Please sign name exactly as it appears above and insert actual date of signing.**

IF THERE IS MORE THAN ONE INVENTOR USE PAGE 2 AND PLACE AN "X" HERE ☒

1 ***Typewritten Full Name of Second Joint Inventor (if any)***

Luc (Given Name) | Middle Initial | VINCENT (Family Name)

2 ****INVENTOR'S SIGNATURE:**

3 ****DATE OF SIGNATURE:** 12 / 30 / 99 (Month / Day / Year)

Residence: Palo Alto (City) | California (State or Province) | USA (Country)

Citizenship: US

Post Office Address: (Insert complete mailing address, including country)

4237 Manuela Avenue

Palo Alto, CA 94306

1 ***Typewritten Full Name of Third Joint Inventor (if any)***

Given Name | Middle Initial | Family Name

2 ****INVENTOR'S SIGNATURE:**

3 ****DATE OF SIGNATURE:**

Month | Day | Year

Residence:

City | State or Province | Country

Citizenship:

Post Office Address: (Insert complete mailing address, including country)

1 ***Typewritten Full Name of Fourth Joint Inventor (if any)***

Given Name | Middle Initial | Family Name

2 ****INVENTOR'S SIGNATURE:**

3 ****DATE OF SIGNATURE:**

Month | Day | Year

Residence:

City | State or Province | Country

Citizenship:

Post Office Address: (Insert complete mailing address, including country)

1 ***Typewritten Full Name of Fifth Joint Inventor (if any)***

Given Name | Middle Initial | Family Name

2 ****INVENTOR'S SIGNATURE:**

3 ****DATE OF SIGNATURE:**

Month | Day | Year

Residence:

City | State or Province | Country

Citizenship:

Post Office Address: (Insert complete mailing address, including country)

****Note to Inventors: Please sign name exactly as it appears and insert the actual date of signing.**
This form may be executed only when attached to the first page of the Declaration and Power of Attorney form and the specification (including claims) of the application to which it pertains.